DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

Representations and Processes in Decision Modelling

S. Greenhill, S. Venkatesh,
A. Pearce and T.C. Ly

DSTO-GD-0318

Representations and Processes in Decision Modelling

Stewart Greenhill, Svetha Venkatesh, Adrian Pearce

Curtin University of Technology

Thanh Chi Ly

Maritime Operation Division

Aeronautical and Maritime Research Laboratory

DSTO–GD–0318

ABSTRACT

This report contains a survey by Curtin University on decision modelling which covers:

1. Our current understanding of how we make decisions, and points out our qualities, our weaknesses and the types of aids that could help us. Of note is the theory that people commit to options even though alternatives exist once a situation has been recognised.
2. Techniques useful for eliciting and representing knowledge about how experts make decisions.
3. What is situation assessment, and how others have tried to capture the process and use the captured information.
4. The different technologies that could be employed to represent the process of situation assessment.

This report represents the first step of a larger project to represent how submarine commanders assess situations.

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION DSTO

Published by

DSTO Aeronautical and Maritime Research Laboratory
506 Lorimer St,
Fishermans Bend, Victoria, Australia 3207

Telephone: (03) 9626 7000
Facsimile: (03) 9626 7999

AR No. AR 012-146
March, 2002

APPROVED FOR PUBLIC RELEASE

Representations and Processes in Decision Modelling

EXECUTIVE SUMMARY

There is a growing need to improve operational analysis models used to support the design, acquisition and implementation of submarine combat systems, including the decision support systems that open combat system architectures are making easier to implement. Our research is directed towards improving the representation of the command decision process, which is the weak link in current models of submarine operations. Our approach is to make situation assessment explicit in the submarine model and to separate situation assessment from the generation of tactical responses. A formal language is used to describe situations in the domain of submarine operations, including mission and tactical goals. The situation assessment processor binds information about the situation, produced in this case by a simulation, to symbols in the formal Situation Description Language taking into account the situational context and spatio-temporal relationships in the information. The result is a textual and graphical description of the assessed situation. This report contains a survey by staff of Curtin University of decision modelling. It provides the necessary background prior to developing the Situation Description Language.

Situation Assessment is a "Multi-perspective multi-membership hierachical pattern recognition problem". Faulty situation assessment will lead to poor decisions. For instance, 175 military aviation mishaps, and 88 percent of major aircraft accidents were attributed to poor situation asessment. The domain experts know how to assess situations. To investigate the process we must elicit the know-how and be able to store it. There is no all encompassing technique for knowledge elicitation. It is suggested that knowledge representation and elicitation processes should:

1. Be compatible with human cognitive processes;
2. Elicit from a group of experts at the same time where possible;
3. Efficiently integrate multiple expert views from different sessions;
4. Elicit from written sources where possible;
5. Minimise burden on experts;
6. Use a knowledge representation that permits easy update, and
7. Use computational efficiency to minimize communication between machine and user.

When people make decisions it is not on the basis of actual data, but on internal representation or perception of that data. In addition, people structure their decision process with goals, which are specifications of the desired situation. We "chunk" information, where complex patterns may be reduced to single symbol (or chunks) when existing knowledge is taken into account. Our decision-making process is governed by our limitations which include:

1. Limited working memory;
2. Slow cognitive operation;
3. Retrieval of information is biased to information that is recent, frequently recalled, or relates to information currently active;
4. Slow and error prone when dealing with numerical operations; and
5. Poor projection in time and space.

An interesting model of human cognition is the Recognition-Primed Decision model which explains the observation that:

> "... experts rarely report considering more than one option. Instead, their ability to handle decision points appears to depend on their skill at recognising situations as typical and familiar. This recognition suggests feasible goals, sensitises the decision-maker to important cues, provides an understanding of the causal dynamics associated with a decision problem, suggests promising courses of action, and generates expectancies."

The model also accounts for how people commit to options even though alternatives exist once a situation has been recognised. The application of this model has improved decision making processes and training in civilian and military organisations in the US.

The focus of the research is improving representation of the situation assessment process for the purpose of further analysis. A better understanding will lead to improvement in the process, and a potential command decision aid. A number of such aids have been developed by groups in the U.K. and U.S.A.

Contents

Figures

Tables

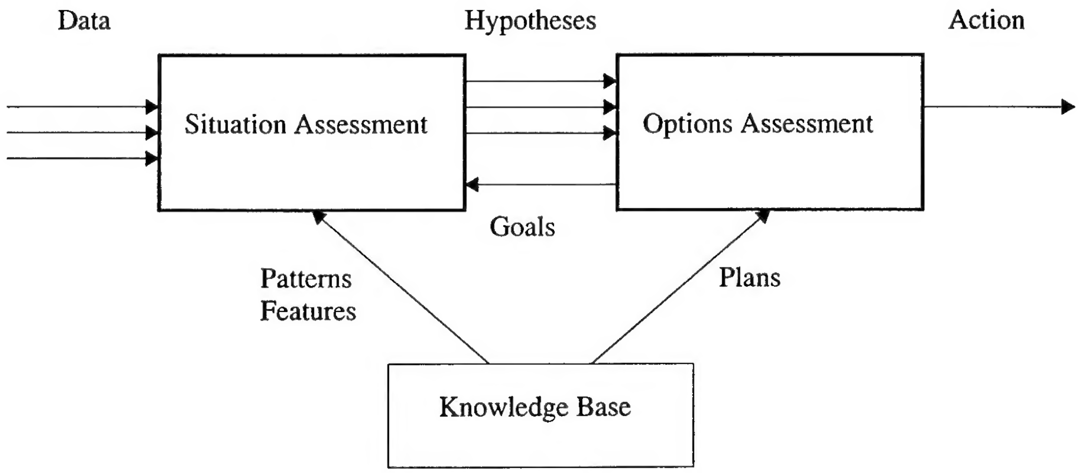


Figure 1: Simple model of a decision process

1 Overview

This document presents a survey of techniques that may be used in a situation assessment processor. The aim is to identify both *representations* and *processes* that may be important in *decision modelling.*

Situation Assessment (SA) is an important part of the decision-making process. Cognitive studies have identified a number of levels at which information is processed in human task performance. At the lowest level, sensory information can be used in a raw state in real-time coordination of movements. At higher levels, judgements are made using abstract models of the environment. It is the task of situation assessment to construct high-level, meaningful models of the environment from low-level sense data. Such models must be constructed in a way that is relevant to the performance of tasks such as reasoning and judgement.

Figure 1 shows a simplified model of a decision process. The Situation Assessment (SA) sub-process interprets sensory data and produces a number of hypotheses, which represent possible interpretations of the situation. To interpret possibly incomplete sensory data, it needs knowledge of the expected properties of objects, including their behaviour. To generate relevant hypotheses, it needs information of overall goals. The goals sensitise the situation assessment process to particular cues or interpretations. The Options Assessment (OA) sub-process examines the hypotheses and its goals and determines what actions to take. A major question for situation assessment is how the required information inputs (sensory data, goals), outputs (hypotheses) and knowledge are represented. For an automatic system, representations are constrained by the fact that knowledge must be elicited from a human decision-maker. This document examines some of the relevant issues.

Section 2 examines models for human task performance. If we are attempting to represent some aspects of human judgement, it is necessary to understand how humans process and represent information. It is important to design a "compatible" knowledge representation that preserves the original meaning of the knowledge. This is also important

for knowledge elicitation; there is no point using a highly statistical representation (for example) because humans are poor at statistical judgement and an expert will not be able to express knowledge in the form of statistical dependencies. Both cognitive and computer-based task models are examined; issues in the representation and elicitation of knowledge are identified.

Section 3 examines existing models of situation assessment. Some of these attempt only to give a system-level description of situation assessment process. They examine what information is involved, how is it acquired and processed. Others describe actual implementations of situation assessment systems (primarily in the naval domain). While detailed technical information is not available, many relevant issues are discussed.

Section 4 discusses issues relevant to situation assessment and the representation of tactics in the submarine domain. A number of existing systems are described.

The remaining sections deal with general areas that are relevant to the representation of situational and tactical knowledge. Section 5 describes issues in temporal reasoning. Section 6 describes issues in spatial reasoning. Section 7 discusses Coloured Petri Nets. These can be used to model interacting processes in a way that enables certain properties to be formally verified. They are an important tool in the study of complex systems, including command and control systems. Section 8 discusses representations of uncertainty. This is important in situation assessment problems since both the knowledge and data will have elements of uncertainty to be represented and dealt with.

2 Models of Human Task Performance

> "The most fundamental requirement for any knowledge elicitation technique is compatibility with the knowledge that a human expert can provide " [4]

In many areas of human endeavour there is a significant interest in improving the quality of human performance at decision-making tasks. There are several approaches to this problem.

The first is the development of *Decision Support Systems* (DSS) to assist humans in decision-making tasks. These systems may work in many different ways, but their essential task is to enable the decision-maker to overcome their own *cognitive limitations* (see Section 2.1).

A second approach is the development of instructional systems to improve the acquisition of expert level decision-making skills. Experience is a vital part of learning. The use of intelligent simulations can provide experience in environments that would not be attainable under normal conditions. (eg. flight simulators).

A third approach is the development of systems that automate decision-making tasks using elements of human reasoning processes. These systems can be deployed to:

- Free the human from certain tasks so that they may concentrate on others (eg. autopilots). This is particularly important in tasks where human performance is affected by factors like fatigue and limitations on attention.

- Behave in a human-like manner within a simulation (eg. computer-generated forces in military simulations).
- Provide reference decisions or explanations against which a human decision-maker can evaluate their own decisions or interpretations (eg. medical diagnosis systems).

Each of these approaches requires a representation of human task performance. Often this involves understanding how individual human experts perform these tasks. In practice, this means studying the general knowledge, specific knowledge and reasoning processes of experts in order to create a model of the task that exhibits some of the behaviour of the expert.

This section examines the problem of eliciting and modelling the knowledge that governs human task performance. In designing any knowledge-based system it is essential to establish what kind of information is to be represented and how this is to be gathered and ultimately used. Where human reasoning processes are involved, these factors are all affected by idiosyncrasies of human cognition. Some of the important considerations are described below.

The human information processor is faced with a number of fundamental limitations which affect the way that decisions are made (see 2.1). This results in the use of heuristics, mental models, abstraction, and model transformation (analogical reasoning). These factors must be considered when designing tasks and decision support systems. According to a popular model (see 2.2) human decision-makers use a hierarchy of knowledge representations. It is important to understand how these interact in the decision process in order to elicit knowledge from an expert. Some forms of knowledge (eg. explicit, procedural, rule-based knowledge) are relatively easy to express. Other forms of knowledge (eg. implicit, perceptual, and higher-level knowledge) are often used without conscious control or awareness and must be probed using structured elicitation techniques (see 2.3). Expert decision-makers rarely report considering more than one option. According to the RPD model, this is becuase a large amount of option generation and evaluation happens without conscious awareness within the situation assessment process.

A number of existing computer-based task modelling systems are examined. The COGNET framework represents knowledge about tasks and the changes in attention between them. COGNET task models are elicited through structured interview techniques and can be used to predict behaviour in human-computer interactions. This makes them useful both for simulating human-like behaviour, and for designing intelligent user-interfaces (see 2.4). Procedural reasoning systems model the effects of tasks (ie. WHAT they achieve) as well as their structure (ie. HOW to perform them). These systems can be used to perform limited planning based on goals (see 2.5). A number of graphical systems exist to enable experts to describe procedural tasks. These include NASA's TARGET system (see 2.7) and a schema-based approach designed by the U.S. Naval War College (see 2.6). These systems concentrate on how to perform tasks, and are therefore not useful for reasoning. However, it is claimed that the latter system can be used for situation assessment.

2.1 Human Cognitive Limitations

Research into decision-making has resulted in a substantial body of knowledge on human decision making [57]. This section summarises some of the features and limitations of human decision-making. It also identifies some areas where appropriate technology can be used to overcome these limitations. A detailed protocol for the design of Decision Support Systems (DSS) is outlined by Zachary [57].

Three attributes are important in understanding human decision making:

1. *Uses of internal (mental) representations.* When people make decisions, it is not on the basis of actual data, but on internal representations or perceptions of that data. A decision maker who has a deep understanding of a problem will have a more complete representation of the problem. An expert knows what to look for and how to interpret data. A novice might not now how to interpret the same data. People strongly organise decision processes around their internal representation of the problem. If this is incomplete or incorrect, it can lead a person to misinterpret or ignore important data that does not 'fit in'.

2. *Pursuit of goals.* People structure their decision process with goals, which are specifications of desired situations. The basis for a decision will thus be something like "what do I want to happen?"

3. *Chunking of information.* What a person perceives depends on what the person 'already knows.' People interpret the world in terms of units of meaning, not in units of information as defined in the 'information theory' sense. Complex patterns may be reduced to single symbols (or chunks) when existing knowledge is taken into account. A decision-maker will perceive information in 'chunks' that are consistent with the internal representation more easily than information that is inconsistent with it.

There are five general human information processing limitations that constrain decision processes:

1. *Working Memory:* People process information mentally in a form that is intermediate between purely perceptual and long-term memory; this intermediate form is working memory (WM). Only information in WM can be reasoned about, or used to make a decision. Studies indicate that it is generally possible to hold only between three and nine chunks of information within WM. This limits the number of items that a decision-maker can deal with simultaneously, even if more items are available or required. Information in WM decays rapidly unless actively rehearsed; estimates range from 7 to 13 seconds per chunk.

2. *Speed of Cognitive Operations:* People cannot reason instantaneously. Each reasoning operation (comparing two data, making an association with a past event, making an inference from a datum to a hypothesis) requires a fixed amount of time (about 0.1 second). Complex reasoning processes require much more time because they involve a string of elementary reasoning steps. This is important in real-time situations where little time may be available for deliberation.

3. *Retrieval of Information:* There are two sources of information available to the human decision-maker: sensory data and information from past experiences. The ability to recall specific information from long-term memory is limited. People demonstrate clear biases for recalling information, favouring more recently learned information, more frequently recalled or rehearsed information, and information more semantically related to information currently active in WM.

4. *Numerical Operations:* Without computational tools, people perform numerical operations with relative difficulty. Each operation takes much more than the 0.1 second minimum, and there are frequent errors and instances of forgetting and recalculating. People are also aware of this limitation, and often seek to avoid decision processes involving a large amount of computation, instead favouring qualitative or heuristic reasoning in decision processes. This presents problems with computer-based aids that are based on numerical approaches. The process of interpreting numbers can slow a person down, or even lead to ignoring a numerical approaches in favour of more comfortable qualitative approaches (eg. [42]). Thus, it is important in such systems to provide appropriate visualisation aids.

5. *Projection in time and space:* People often use visual imagery as decision problem representations. Unfortunately, people are not as good at the projection process as they believe they are. Thus, they may represent the trajectories of two vehicles as curves on a plane, but will be unable to accurately project the interception point without first drawing it on a piece of paper. The same limitation applies to projecting physical processes in time. A person may watch a moving symbol on a screen, but will not be able to project its location in 5 seconds with any accuracy.

Zachary [57] argues that there are only six general decision-support needs that are both *common* and *have known forms of computational support.* These arise from the previously described problems in the human cognition process. They are:

- *Inability to predict processes:* People have difficulty projecting real-world processes forward in time. This is particularly true when uncertainty is involved. Many decision makers may feel that it is impossible to predict a complex process "in the head" and so do not try, relying instead on general purpose heuristics and knowledge of "average cases". Computers can predict well-understood processes numerically.

- *Difficulty in combining competing attributes or objectives:* In many situations, there are several attributes that can describe an expected outcome of a decision. Often, criteria-combination rules are ill-defined. In these cases, human decision-makers can benefit from access to the knowledge of expert decision-makers, which can be provided through computerised knowledge-bases.

- *Inability to manage information needed in the decision process:* Decision makers often fail to make use of all the information available to them simply because they are unable to manage it effectively "in their head". A human decision-maker can easily be overwhelmed by a large amount of information and as a result can fail to process key inputs or fail to recall and apply crucial pieces of knowledge. Decision makers may be aware of this problem and so may not even try, relying on heuristics.

Computers can be used to organise and access information to assist the human decision-maker.

- *Problems in analysing or reasoning about the situation:* Human decision-makers often know how they would like to think about a problem, but are unable to do so because of time or mental resource limitations. In many cases, the problems that humans find difficult can be solved by existing computational algorithms.
- *Difficulties in visualising:* People tend to use visual representation in decision processes, but frequently have difficulties in manipulating these representations, particularly in a quantitative manner. People are much better able to make quantitative projections with their visual representation if they can deal with a concrete (ie. explicitly drawn) representation, rather than a purely mental one. Computers can assist in such problems by providing appropriate graphical representations.
- *Quantitative inaccuracies in heuristic judgements:* Often, human decision-makers are required to make heuristic judgements. Skilled decision-makers can make such judgements wil high reliability and consistency, but when they have a numeric aspect there is often a systematic bias or noise in the judgements. While computers cannot rival the human judgement ability, they can be used to remove some of the noise and bias.

2.2 Skills, Rules and Knowledge

Rasmussen [44, 45] distinguishes three categories of human behaviour according to different ways in which humans represent knowledge about an environment or system. Depending on the task, perception, action and information processing may occur at different levels in a hierarchy of cognitive control. This hierarchy is depicted in Figure 2.

At the lowest level, *skill-based behaviour* represents sensory-motor performance, based on real-time multivariable and synchronous coordination of physical movement with a dynamic environment. Following a statement of intention, these usually take place without conscious control as smooth, automated, and highly integrated patterns of behaviour. The flexibility of skilled performance is due to human abilities to compose and adapt from a large repertoire of prototypical movement patterns those that are suitable for a particular purpose. These patterns are activated and chained by cues and require no choice among alternatives.

At the next level, *rule-based behaviour* is the composition of sequences of actions controlled by stored *rules* or *procedures.* These procedures may have been derived empirically during previous experience, communicated from another person in the form of an instruction or recipe. If rules are not available, they may need to be prepared by conscious problem-solving and planning. Performance of rule-based behaviour is goal-oriented, but structured by "feed-forward control." Often the goal is not even explicitly formulated, but is found implicitly in the situation (or *context*) that activated the stored rule. In general, rule-based behaviour requires a conscious preparation of the procedure or sequence beforehand. Attention will look ahead to identify the need for rules and actions in the near future, and it will look back to recollect useful rules from past encounters. Skill-based

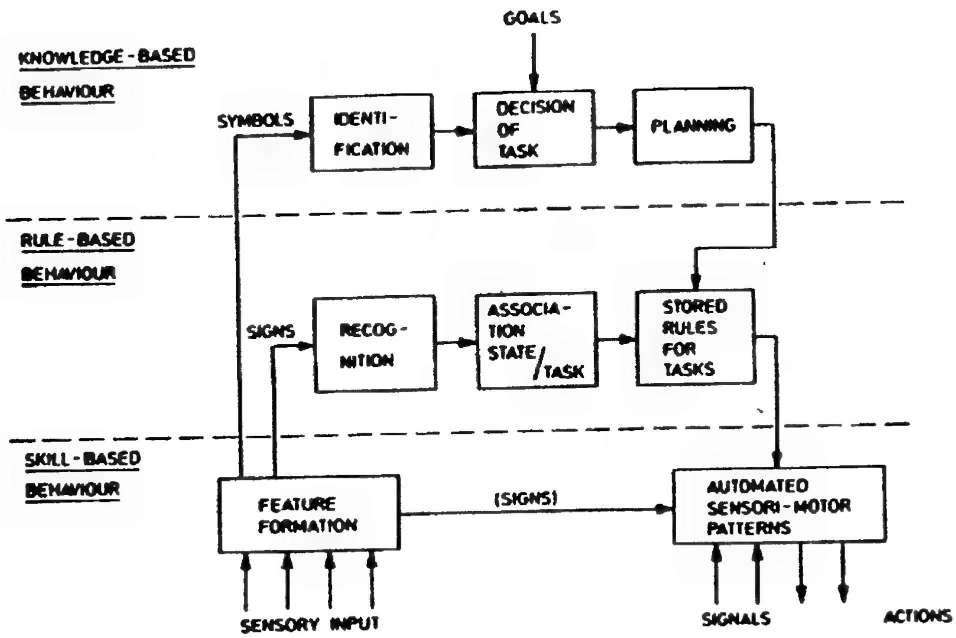


Figure 2: Hierarchy of performance levels for skilled human operators [44]

performance proceeds without conscious attention, and a person will be unable to describe how the task is performed and what information is involved. Rule-based performance is generally based on explicit know-how, and the rules can be reported by the person.

During unfamiliar situations, faced with an environment for which no rules are available from previous experience, the control of performance must move to a higher conceptual level, in which performance is goal-controlled and *knowledge-based.* In these situations, goals are explicitly formulated, based on an analysis of the environment and the overall aims of the person. Then a useful plan is developed by selection. Different plans are considered and their effect is tested against the goal. This testing may be done physically by trial and error, or conceptually by understanding the environment and considering the effects of the plan. At this level of reasoning, the problem domain is explicitly represented by a "mental model" which may take several different forms.

The human data-processor is faced with several problems in modelling the causal and functional properties of the environment. Only a few elements of a problem can be within the span of attention simultaneously (see Section 2.1). This means that the complex net of causal relations of an environment must be treated in a chain of mental operations, often leading to effects like the "law of least resistance" and the "point of no return". That is, strategies that depend on sequences of simple operations are intuitively preferred, and little tendency will exist to pause in a line of reasoning to backtrack and develop alternative parallel paths.

An effective way to counteract limitations of attention seems to be to adapt the mental model of the environment to fit to the specific task in a way that optimises the transfer of previous results and minimises the need for new information. The efficiency of human cog-

nitive processes seems to depend upon an extensive use of *model transformations* together with a simultaneous updating of the mental models in all categories with new information. This updating may be performed below the level of conscious attention and control.

From analysis of verbal protocols, it appears that several strategies for model transformation are generally used. These include:

- *Aggregation:* Elements of the model are aggregated into larger chunks within the same model category as familiarity with the context increases.
- *Abstraction:* The representation of the properties of a system is transferred to a model category at a higher level of abstraction.
- *Analogies:* The representation is transferred to a category of model for which a solution is already known, or for which rules are available to generate the solution.

Tasks are often analysed in terms of sequences of separate acts. In general, several functions are active at the same time (see Table 1). However, attention is scanning across time and activities in order to analyze past performance, monitor current activity and plan for foreseeable future requirements. This prepares the dynamic internal mental world model for oncoming demands. It also allows cues and rules to be rehearsed and modified to match requirements. Symbolic reasoning is used to understand responses from the environment, and prepare rules for foreseeable situations that may be unfamiliar. Attention may not always be focussed on current activities: different levels may simultaneously be involved in the control of different tasks in a sequential or parallel processing mode.

2.3 Recognition-Primed Decision (RPD) Model

The critical decision method (CDM) is a knowledge elicitation strategy developed by Klein [23]. This method was developed to elicit *tacit* and *perceptual* knowledge in addition to the *explicit* knowledge that is part of most expert-system engineering problems. It is defined in the context of the recognition-primed decision (RPD) model, which has been developed to describe a class of expert decision-making behaviour.

Klein describes expertise as a combination of:

- Explicit, objective knowledge. This includes *factual knowledge*, *if/then rules*, and *analytical procedures*. This knowledge is easily used by knowledge-engineers in building expert systems since it can be expressed directly.
- Tacit knowledge, which is resistant to being articulated. This includes appreciation of *contextual* implications. Context includes the background information that enable experts to articulate rules and apply procedures. *Analogical reasoning* often involves tacit references to the way that a situation is recognised.
- Perceptual learning, which involves the development of a sensory-motor *feel*. These skills often become automatic (such as driving a car, or playing tennis).

Mode	Mental Functions	Temporal Characteristics	Related world
Knowledge based	Planning in terms of functional reasoning by means of symbolic model	Achronic, that is, temporal scale is not maintained in causal reasoning	As can be
Rule Based	Planning in terms of recall of past and rehearsal of future, predicted scenarios	Diachronic, that is, temporal scale is maintained but not synchronised	As has been and may be
	Attention on cue classification and choice of action alternatives	Synchronic, that is, operation in the actual time slot, but not synchronous	As is
Skill Based	Data-driven chaining of subroutines with interrupt to conscious, rule-based choice in case of ambiguity or deviation from current state of the internal world model	Synchronous with real world, operation in 'real time'	As is

Table 1: Interaction between levels of cognitive control [45]

The tendency has been to emphasise objective knowledge, whereas other aspects of expertise are important, if not *essential*, to achieving proficiency. Knowledge elicitation techniques must include some means of representing *tacit knowledge* and *perceptual learning.*

An important finding of research into decision making is [23]:

> ... experts rarely report considering more than one option. Instead, their ability to handle decision points appears to depend on their skill at recognising situations as typical and familiar. This recognition suggests feasible goals, sensitises the decision-maker to important cues, provides an understanding of the causal dynamics associated with a decision problem, suggests promising courses of action, and generates expectancies.

The *Recognition-primed decision* (RPD) model (depicted in Figure 3) incorporates these observations. It assumes that an acceptable course of action may be chosen without conscious generation and evaluation of alternatives. Thus, the emphasis in the model is on *situational awareness.* The model focusses on how people commit to options even though alternatives exist. Once a situation has been recognised:

- We have a basis for expecting certain things to occur but not others.
- We pay attention to cues relevant to the situation
- We have some understanding of what goals are reasonable to achieve.
- We know which types of actions are likely to succeed.

The RPD model has been tested and has been supported by various research teams working in a variety of decision-making settings [22]. It is termed a *singular* strategy, since it involves considering only a single option at a time. It was found that people use recognition-primed decisions in the following situations:

- When *time pressure* is greater, evaluating one option at a time until an acceptable one is found. The reason is that it takes too much time to lay out the alternatives and analyse the evaluation criteria.
- When people are more *experienced* in the domain. With experience, people can be more confident in their ability to evaluate the situation and recognise plausible courses of action as the first ones they consider.
- When the conditions are more *dynamic.* The time and effort needed to set up an analysis can be rendered useless when the context changes.
- When the goals are *ill-defined.* Ambiguity makes it hard to come up with evaluation criteria that apply across all options.

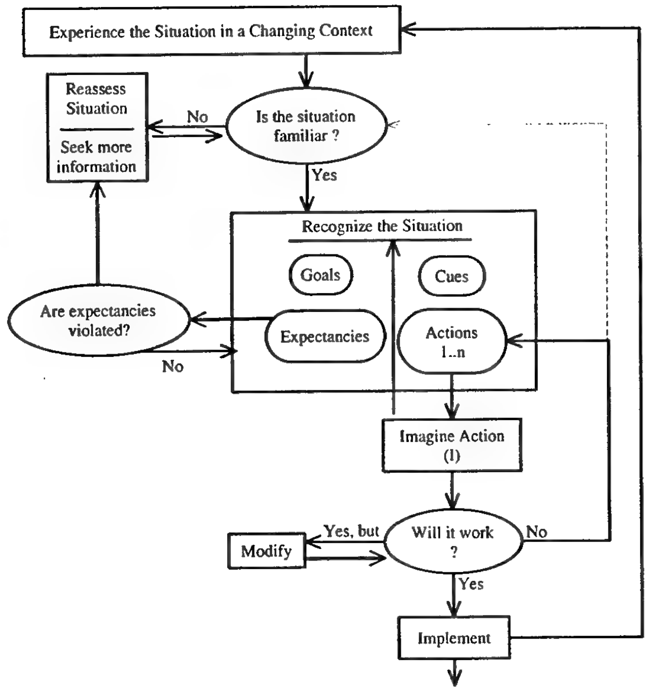


Figure 3: Recognition-primed decision model [23]

In contrast to the singular decision strategy is that of *comparative evaluation* or *rational choice*, where two or more options are compared on the basis of some selection criteria. Until the development of RPD models, comparative models were assumed to be the basis of decision-making. In the above situations, they do not apply. However, people are more likely to use comparative evaluation in these situations:

- When they have to *justify* their choice. Higher authorities usually look for evidence that alternatives were considered.
- When *conflict resolution* is a factor. Here, the interests and priorities of different stakeholders needs to be considered explicitly.
- When the decision maker is trying to *optimise*, finding the best possible course of action.
- When the situation is *computationally complex*. Humans are notoriously poor at computation. Where analysis is required, recognition is generally not possible.

The RPD model has implications for both system design and training. These issues are discussed in detail by Klein [22]. Training in analytical methods may be useful to inexperienced decision-makers. However, experts in many different fields learn best by:

- Engaging in deliberate practice, so that each opportunity for practice has a goal and evaluation criteria.

- Compiling an extensive experience bank.
- Obtaining feedback that is accurate, diagnostic, and reasonably timely.
- Enriching their experiences by reviewing proper experiences to derive new insights and lessons from mistakes.

The analysis of *decision requirements* is important for the design of software systems. For a given task, the decision requirements are the key decisions and how they are made (ie. what decisions are made, what cues are involved, what information and strategies are used). By identifying decision requirements of U.S. Air Force weapons directors in AWACS (airborne warning and command system) missions, simple modifications were suggested for the interface with their workstations. After implementing these changes, a 15 to 20 percent improvement in performance was obtained. Such an improvement would have been very costly to achieve in other ways (eg. by developing faster computers or providing more weapons director training)

2.3.1 Critical Decision Method (CDM)

The CDM is a retrospective interview strategy that applies a set of *cognitive probes* to actual nonroutine incidents that required expert judgement in decision-making. Such nonroutine or difficult incidents are usually the richest source of data about the capabilities of highly skilled personnel. Once an incident has been selected, the interviewer asks for a brief description. Then a semistructured format is used to probe different aspects of the decision-making process. Typically, interviews are tape recorded so that transcripts can be made.

The critical decision interviews consist of the following steps:

Step 1 *Select Incident:* Incidents are selected that can illustrate nonroutine aspects of a domain. The decision-maker is asked to select an incident that was challenging and that, in his or her decision-making, might have differed from someone with less experience.

Step 2 *Obtain Unstructured Incident Account:* The officer is asked to describe the incident. The account proceeds without interruption from the interviewer, except for minor points of clarification. This serves to create an understanding of context on the part of the interviewer. It also serves to activate the officer's memory of the event as a context for questioning.

Step 3 *Construct Incident Timeline:* After the incident has been related, the interviewer reconstructs the account in the form of a timeline that establishes the sequence and duration of each event reported by the officer. Events include both objectively verifiable occurrences and thoughts and perceptions reported by the officer.

Step 4 *Decision Point Identification:* During the timeline construction, specific decisions are identified for further probing. In some cases, verbal cues marking a decision are obvious (eg. "I had to decide whether ..."). In other cases, it may be clear

Probe Type	Probe Content
Cues	What were you seeing, hearing, smelling ...?
Knowledge	What information did you use in making this decision, and how was it obtained?
Analogues	Were you reminded of any previous experience?
Goals	What were your specific goals at this time?
Options	What other courses of action were considered by or available to you?
Basis	How was this option selected / other options rejected? What rule was being followed?
Experience	What specific training or experience was necessary or helpful in making this decision?
Aiding	If the decision was not the best, what training, knowledge, or information would have helped?
Time Pressure	How much time pressure was involved in making this decision?
Situation Assessment	Imagine that you were asked to describe the situation to a relief officer at this point, how would you summarise the situation?
Hypotheticals	If a key feature of the situation had been different, what difference would it have made in your decision?

Table 2: Critical Decision Interview Probes [23]

that an officer has taken one of several possible courses of action, or has made a judgement that affected the outcome, but the officer does not give a clear indication that a decision is being made. Such points are probed if the officer agrees that other reasonable courses of action are possible, or that another officer might choose differently.

Step 5 *Decision Point Probing:* Table 2 summarises the types of probes that have been routinely used. Questions to elicit the details of cue usage are almost always asked first at part of the timeline construction. Goals are an important part of situation assessment, but it is important to elicit specific goals rather than general higher-level goals. Specific goals have reasonably stated alternatives. Probes about options are asked for each decision, both actual and hypothetical.

Critical decision protocols can be coded in a number of ways, but the adopted technique depends on its inteded use. Examples are:

- *Descriptive Decision Model.* Klein describes a study of decision-making amongst fireground commanders involving various degrees of time pressure. Decision points were coded to distinguish the decision strategies employed by these commanders. Factors included whether concurrent or serial (RPD) evaluation was primarily used, whether deliberation was about the situation (situation assessment) or the reaction (option assessment) was required.

- *Critical Cue Inventory (CCI).* This is a collection of all of the informational and perceptual cues that are pinpointed in the protocols. The CCI has been most directly useful in the study of paramedics. Here, the CCI consisted of the cues actually used by paramedics and others to recognise heart attack victims during and prior to their showing of standard symptoms.

- *Situation Assessment Record (SAR).* Since the RPD model treats decision-making as a form of complex pattern matching, much of the expertise elicited appears as situational assessment. For each decision point, critical cues and goals are probed. Often the original situation is maintained throughout an incident, with new information adjusting the original assessment (SA-Elaboration). Occasionally, extreme changes in situation assessment (SA-Shifts) result, causing the decision-maker to modify or replace earlier goals. The SAR records these changes as time progresses.

Klein identifies several practical criteria for judging the effectiveness of knowledge elicitation methods:

- Time needed to apply the methods. It is rare that the knowledge elicitor will have more than two hours at a stretch with a domain expert.

- Cost-effectiveness of data collection and analysis. It is important to restrict the type of data collected to that which can be analysed.

- Timeliness of results. In the development of expert systems it is useful to be able to quickly program and evaluate the results.

- The level of training needed for the knowledge elicitors. For most projects, highly trained elicitors are necessary. It is possible to use personnel with less training by providing a highly structured interview procedure. This requires that the initial procedures be designed by domain experts. The results of subsequent elicitation are thus less informative.

- The packaging of the knowledge elicitation results. This must be presented in a usable form. In the case of expert system development, this is usually a software program.

2.4 The COGNET Model

The COGNET framework [58, 61] is designed for representing real-time task performance. Early publications on COGNET emphasise its use in modelling human-computer interaction (HCI) [58]. However, it has also been applied to the representation of general problems involving procedural (rule-based, in the SRK context) knowledge. One relevant example is in the field of Anti-submarine warfare (ASW) [53, 63, 62]. COGNET task models can be implemented (ie. simulated) on a blackboard architecture known as BATON [60]. This allows the construction of intelligent user interfaces that include "embedded task models" to represent both the task and the user. Section 2.4.1 describes this in more detail.

COGNET assumes that a human decision-maker is performing a number of tasks in a weakly concurrent manner. Each task may be partly completed, interrupted by some other tasks, and perhaps later resumed at the place where it was interrupted. The tasks are interrelated in that they contribute to a high-level problem-solving goal, but only one can be actively pursued at any one time. Thus the *attention* of the human is shared, in real-time, between lower-level tasks.

In a COGNET model, each task is represented using a cognitive task analysis language derived from GOMS [7]. A GOMS model consists of four components: Goals, Operators, Methods, and Selection rules:

- *Goals:* are states of the world that a person is trying to bring about. Goals may refer to physical or cognitive conditions. Goals are hierarchically decomposed into lower-level goals and actions (which may be either *operators* or *methods*).
- *Operators:* are elementary perceptual cognitive and motor actions which a person may undertake to achieve a goal. An operator may be used in many different goals. In HCI environments, operators often correspond directly to a command within a computer interface.
- *Methods:* are compositions of goal/subgoal/operator sequences. A method represents a partial or complete procedure for performing some task.
- *Selection Rules:* are if-then rules for selecting among alternative methods.

COGNET extends the GOMS model by adding [58]

- *An Object Language:*
- *New GOMS Operators:*
- *Triggers:*
- *Suspensions:*
- *Perform Constructs:*
- *Perceptual Demons:*

COGNET has been applied to dynamic tactical environments such as Anti-submarine warfare [53]. Here, operators must continuously balance goals for safety with mission goals and objectives as the tactical situation evolves. These high-level goals introduce specific constraints on operator actions that often conflict with one another. An extension to the basic model allows AND/OR tree diagrams to represent the relationship between high level goals and lower level constraints on operator actions. The goal/constraint trees were modelled for the Submarine Operational Automation System (SOAS) operator using expert knowledge elicitation techniques.

The authors of [53] state that a useful method for eliciting expert knowledge of real-time domains involves using real-time simulations. By providing a context for experts to make

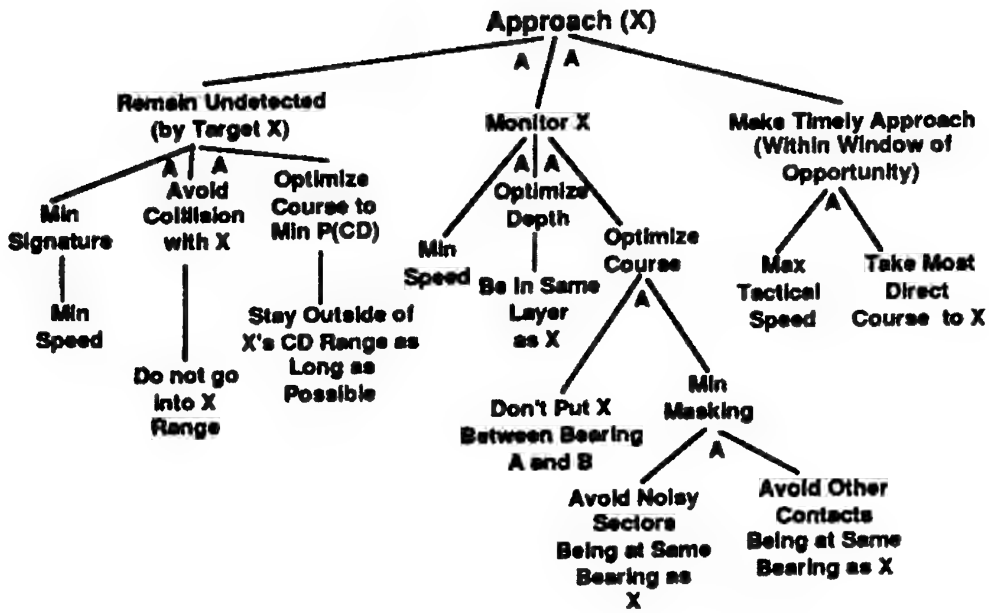


Figure 4: COGNET Goal/Constraint Hierarchy for Approach task [53]

real-time decisions within a controlled environment, simulations dramatically enhance the data collection process. Where such simulations are unavailable, an alternative method is the use of "two player games". This involves creating scenarios in real-time with two domain experts: one to "play" the tactical operator, and the other to "play" the tactical situation. Using the structured knowledge elicitation techniques of Klein (see Section 2.3), data was collected during scenario sessions.

Analysis of the data revealed that the goals differ in terms of how they become activated. Some are active continuously throughout the mission. Others were only activated as the result of a specific set of tactical circumstances. Additional distinction was made between goals that are activated for a particular contact, and those that are more general in nature. To activate contact-dependent goals, the contact for which they are activated is specified. Figure 4 shows a simplified goal hierarchy for the goal "Approach Target (X)". Note that the target name X appears in many of the sub-goals.

During the modelling process, it was observed that operators often, when resolving goal conflicts in order to make planning decisions, simply failed to incorporate all the relevant constraints into the decision. It also appeared that the experts were aware that they were relaxing constraints that they considered to be least important, because they simply could not cognitively keep track of all the constraints relevant to a particular decision. This is in keeping with human cognitive limitations described in section 2.1. This led to the development of a goal-management decision aid that allows operators to keep track of all the constraints that can affect course, speed and depth when determining submarine course.

2.4.1 Embedded Task Models

By embedding a model of human-computer interaction within a user-interface, the quality of interaction is enhanced. Intelligent interfaces can recognise what needs to be done and can take actions such as:

- alerting the operator of the opportunity or need to perform or resume a task.
- offering to perform the task automatically on operator approval using its model of how the task should be performed given the current situation.

An example from the Air ASW domain is described by [60]. Operators in aircraft drop sonobuoys into the water to gain acoustic contact on submarines. As the submarines move, more buoys must be dropped to maintain contact. As the tactical situation progresses, the COGNET task model is able to infer the current situation from perceptual monitors that track events, as well as the actions of the operator. When the situation calls for a new tactic, the intelligent subsystem can recognise the situation and offer to perform the needed task.

A series of trials of the Air ASW model involved subjects solving simulated problems. A COGNET model of the task was found to predict the actions of subjects [46]. Each problem contained an average of 400 user actions and 40000 elemental display events. Separate data were collected from subjects used to build the original model and new (but equally expert) subjects. The model was found to predict 90% of task instances for the original subjects, and 94% for new subjects. For predicted tasks, the model prediction was found to lead the actual task by an average of 3.2 minutes for the original subjects and 2.2 minutes for new subjects (over an average problem duration of 90 minutes).

A set of tools has been developed to facilitate the creation of "Interface Agents" using the COGNET model [59]. GINA (Generator of INterface Agents) allows:

- Creation and modification of COGNET models
- Implementation and debugging of executable software versions of these models as the intelligent kernels of interface agents
- Application of the executable models as fully autonomous intelligent agents.

Several classes of agent are identified:

- *Adaptive Interaction Agents.* This kind of interface agent provides an existing user interface with a means of adapting its information presentation to the context in which it is being used. An example of this application is an adaptive man-machine interface (AMMI) for caution / warning / advisory information on an advanced helicopter. The agent takes no actions of its own. However, it can reason about what the user knows at a point in time, and can provide information that reflects what an 'expert' user would require at that stage in the mission.

- *Task-Oriented Agents.* These agents are given responsibility for a specific set of tasks to support the user. An example is the Cancer Patient Retrieval Agent (CAPRA) which models two types of users: the clinical oncologist and the medical database searcher. The clinical oncologist model examines on-line records as a human clinician is updating them, and identifies features or questions that could be resolved or augmented with database information (eg. new treatments or drugs that might be relevant). The questions are then passed to the database searcher model which identifies the relevant database, connects to it, queries and returns the results which are then presented to the clinician by the oncologist model.

- *Cooperative Agents.* These agents perform autonomous tasks in close coordination with a person or other agent. An agent is currently being developed to support Naval commanders in identifying and making use of all the information that is available to them, and integrating this information into a coherent, comprehensive, and consistent tactical picture. Direct observations of the commander's interaction with his tactical information system display and with his staff are used to acquire or identify specific information needs of the decision maker and support team. The needs are identified along with constraints such as the time available, relative priority, how concrete or broadly focussed a search should be. Other agents search, prepare and present the information in the most appropriate format.

- *Surrogate Users.* These are agents that simulate users within training and simulation systems. In large simulations where there are potentially more roles than players, these agents can act as surrogates for missing human users. The model described by [53] is being adapted to develop a surrogate adversary for a new generation of distributed submarine trainers.

2.5 Procedural Reasoning

Most AI planning systems involve the use of various techniques for searching and transforming state spaces. Much of what humans do in everyday tasks does not involve this type of reasoning. In the SRK sense, it is rule-based rather than knowledge based. The Procedural Reasoning System (PRS) is a system developed by NASA based on a formal semantics of processes [11].

A *process* is an abstract mechanism that can be executed to generate a sequence of world states, called a *behaviour* of the process. The set of all behaviours of a process constitutes the *action* generated by the process. A process can be modelled by two sets of behaviours, one representing the *successful* behaviours, and the other representing the *failed* behaviours.

Each process is represented by a labelled transition network with distinguished start and finish nodes and arcs labelled with subgoal descriptions. Figure 5 shows an example process. A *process assertion* expresses the fact that, under certain conditions, successful execution of the process will result in a certain behaviour being achieved. A process assertion consists of a process description P, a precondition c denoting a set of world states in which the process is applicable, and an effect g characterising the set of successful behaviours that process can generate when commenced in a state satisfying c. The process

assertion is written $c < P > g$. Process assertions may use variables, which can appear in c, P, or g.

Several constructs can be used to specify temporal action descriptions:

- $(!p)$ denotes those behaviours whose last state satisfies p.
- $(?p)$ denotes those behaviours whose first state satisfies p.
- $(\#p)$ denotes those behavious all of whose states satisfy p.

Within PRS, an interpreter explores paths in a process description. To transit an arc, it unifies the arc assertion with the effects of the set of all process descriptions, and executes a set of unifying processes, one at a time, until one terminates satisfactorily. The system consists of a database containing currently known *facts* (or beliefs) about the world, a set of current *goals* (tasks), a set of *process assertions* (plans) that describe procedures for achieving goals or reacting to particular situations, and an *interpreter*. At any moment, the system also has a *process stack* containing all currently active processes. This stack can be viewed as the system's current *intentions*.

The PRS is well suited to applications involving fault diagnosis in complex systems. One such application is the diagnosis of the Reaction Control System (RCS) of NASA's space shuttle. This application is described further by [11].

It is not clear exactly how PRS rules are elicited. Due to the abstract, formal structure of PRS rules it is unlikely that a domain expert could formulate rules without some training. Thus, a knowledge engineer would be required for the knowledge elicitation process.

2.6 Schema-based Knowledge Elicitation

Schemata have been used to structure knowledge for miltary situation assessment problems [35]. Schemata are mental structures that are believed to represent how memory is organised. Example theories include the frame theory of Minsky [32] and the scripts theory of Schank [49].

Different experiments in knowledge modelling have given rise to various types of schemata. However, there are several factors in common. They are defined by a set of slots, by constraints on these slots, and by relationships with other schemata. The slots correspond to the essential features of the entity represented by the schema.

Noble [35] defines situation assessment to include an estimate of the purpose of activities in the observed situation, an understanding of the roles of the participants in these activities, inferences about completed or ongoing activities that cannot be directly observed, and inferences about future activities. An expert system for situation assessment has been developed by the U.S Naval War College. It includes an assessment system and a knowledge elicitation system. The knowledge elicitation tool is based on the assumption that the knowledge used for situation assessment is organised as schemata in the expert's mind. This motivates a knowledge elicitation procedure in which the expert communicates

Precondition:	TRUE
Effect:	(! (DEFEATED $GIANT))

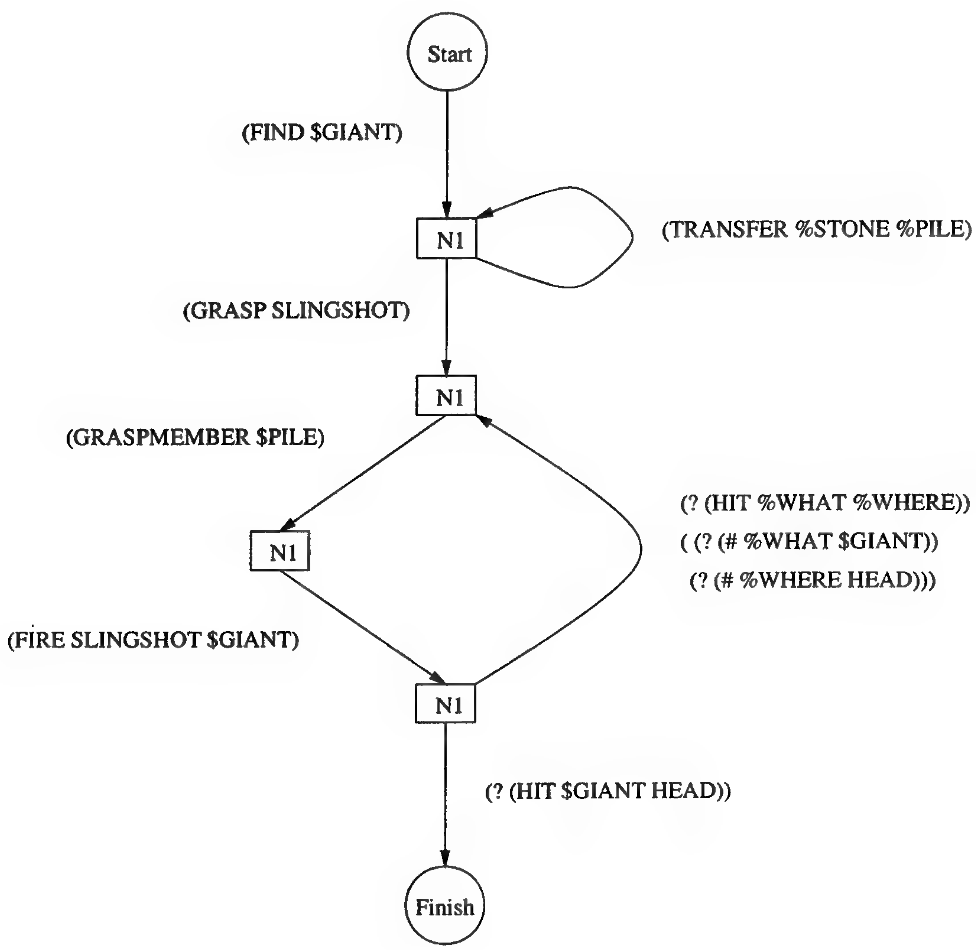


Figure 5: Simple process model in PRS [11]

his understanding of an activity by drawing a diagram, similar to the type used in project planning, that describes how each activity is accomplished.

A knowledge elicitation tool implemented in Pascal for Macintosh computers allows experts to structure schema-based knowledge. The system produces *templates*, which are data-structures that represent schema (schema are mental entities). The tool helps the expert describe the slots in the schema for a hostile activity. These slots are the activity's participants and events. The tool helps the expert set various types of constraints on the slots, including constraints on the characteristics of slot fillers, and constraints between different slots. The tool also allows the expert to describe activity as part of a hierarchy of activities.

Constraints may be *elastic*, which means that they specify the degree to which an observed features is able to fit a template slot. Such constraints can explain how people rate "goodness of fit."

Figure 6 shows an example template containing knowledge about a class of military operations: air strikes against a hostile naval force. The template specifies a most typical attack, as well as the variations among attacks. In the diagram time runs along the horizontal axis, and each row signifies an individual participant. Variability is specified by slot elastic constraints, and by allowing multiple templates to match a slot.

Temporal and logical relationships between slots can be specified using four types of relationships: causes, enables, precedes and accompanies. The operator describes these relationships graphically using a procedure shown in Figure 7. Drawing lines between the margins of slots establishes a default temporal relationship. Times are represented as fuzzy intervals with four components: the smallest possible duration between slots, the smallest typical duration, the largest typical duration and the largest possible duration. These times may be adjusted by the operator.

Experience with the knowledge elicitation tool has been mixed. Users liked the time-event depiction of operations. However, difficulties included the following:

1. Users are uncertain how to partition an activity into events.
2. Users are uncomfortable using the same events in different activities, even though events can occur in many different contexts.
3. Users with different situation assessment specialties tend to represent different aspects of the same data.

2.7 The TARGET Knowledge Elicitation Tool

The TARGET system [47] is a knowledge elicitation tool designed specifically for procedural tasks. At NASA's Johnson Space Centre, over 20 knowledge acquisition tools were evaluated to identify tools that would:

- Elicit or document procedural knowledge
- Run on commonly-available hardware configurations

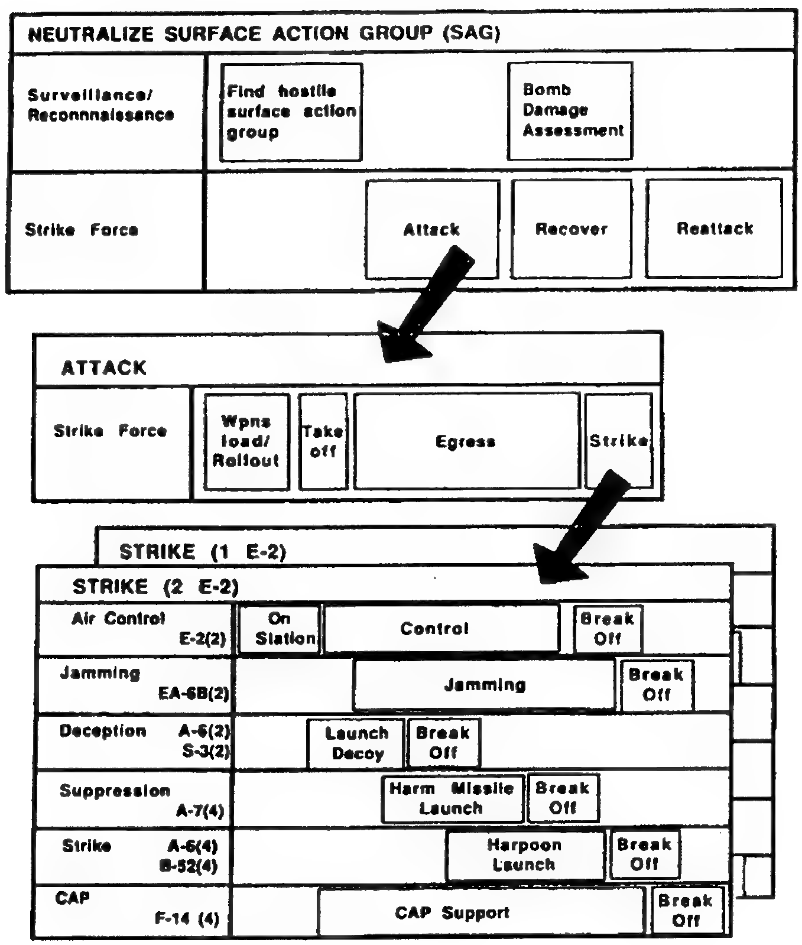


Figure 6: Example template showing hierarchy of activity [35]

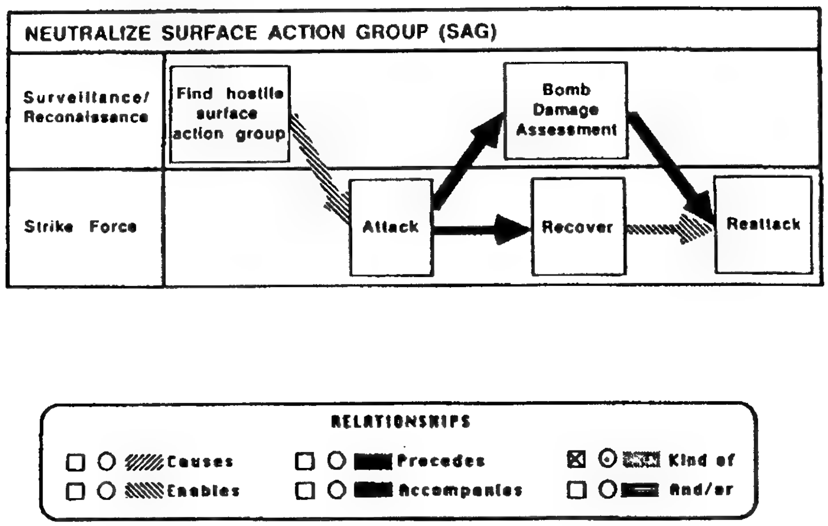


Figure 7: Defining temporal relationships between template slots [35]

- Offer an easy-to-use and robust environment for acquiring procedural knowledge.

The Task Analysis Rule GEneration Tool (TARGET) was designed to meet perceived difficulties with existing systems. TARGET allows users to graphically decompose tasks or procedures using a box-flow presentation style within a windowed environment. It is designed to fit within NASA's existing intelligent computer-aided training systems (ICATs) which use CLIPS as a representation for rules. Thus, TARGET is able to generate CLIPS descriptions of the modelled tasks.

TARGET represents tasks using a hierarchical network of elements. There are four major categories:

- *Required Tasks:* are those that must be performed to complete a process. These are indicated by white rectangles.
- *Optional Tasks:* are those that may be performed, but are not necessary to complete a process. These are indicated by grey rectangles.
- *Decision Tasks:* are linked to a number of different tasks, depending on the result of a decision process. Each outcome is indicated by a labelled arc out of a hexagonal structure.
- *Control Structures:* are used as "go-to" or iteration mechanisms. These are indicated using an ellipse.

Figure 8 shows a sample task layout using some of the elements described. Tasks that are displayed as a layered stack have been hierarchically decomposed into sub-tasks.

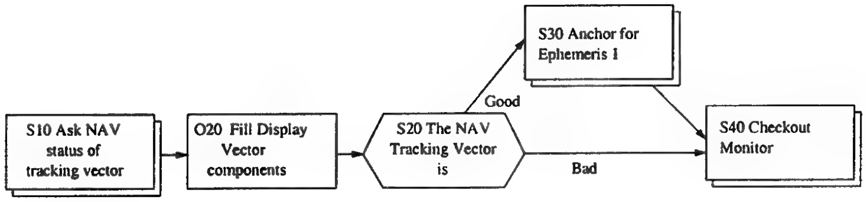


Figure 8: TARGET graphical task report [47]

For procedural knowledge, TARGET simplifies the task of knowledge acquisition by giving the expert the ability to visualise and organise a task description. It should be noted, however, that TARGET's procedural task elements are very simple compared to those described in Sections 2.6 and 2.5. This limits the types of tasks that can be modelled.

3 Issues in Situation Assessment

In a review of 175 military aviation mishaps, situation assessment was found to be the leading causal factor. Situation assessment was also targeted in 88% of major aircraft accidents involving human error. Pilot decision-making can be improved by providing better situation assessment (SA) to the pilot [16]. Despite much work in the area, it is still difficult for automatic systems to encapsulate human decision-making processes. However, improved situation assessment can help human decision-makers make better decisions.

Situation Assessment (SA) involves the interpretation of elements in the environment (time and space), and the comprehension of their meaning [16]. This section examines issues related to the design of situation assessment systems. Section 3.1 examines two general models for military situation assessment. While these do not propose the use of particular analytical techniques, they identify important features of the problem: the nature of situation assessment, the types of decision problems that are involved, and types of knowledge that are required for these decisions to be made. Three existing implementations of military situation assessment are examined: SATE (see 3.2), SAP (see 3.3), and a system described by Kirillov (see 3.4).

From the described literature, it is clear that there are several important issues.

Firstly, there is a substantial "information gap" between information that is available (inputs to SA) and information that is required (outputs from SA). To form a situation assessment, a complete, high-level interpretation is required. Usually only incomplete, low-level information is actually available. A certain amount of the problem can be handled as a bottom-up process. However, the incomplete nature of the available data requires significant top-down processing (for example, the maintenance of several hypotheses that may be only partly validated by the available data). This means that a significant amount of knowledge about "expected patterns" needs to be known *a priori*. In a sense, this is similar to problems encountered in speech recognition: without high-level semantic knowledge, even the most basic "low-level" operations (determining the boundaries between spoken words) are impossible.

Secondly, in a military environment much of the low-level information is kinematic in nature (objects moving in time and space). We might know the position and velocity of an object, and possibly its type (based on emissions data, sightings, etc). Any further information about the entities must be inferred from their behaviour. For example, knowing the velocity or maneuverability of an entity will constrain possible interpretations of what it is (different types of vehicles have different capabilities). Certain patterns of behaviour will also suggest certain roles or functions (eg. reconnaissance). To encode this knowledge requires a formalism that can describe patterns in time and space (eg. trajectories), as well as the relationships between entities. These relationships may be simply spatial (eg. formations), or they may be functional (eg. one vehicle escorting another, or gathering information for a group).

3.1 General Models for Military Situation Assessment

3.1.1 Ben-Bassat and Freedy, 1982

An important overview of the military situation assessment problem is given by Ben-Bassat and Freedy [4]. They characterise the situation assessment task as being a "multi-perspective multi-membership hierarchical pattern recognition problem".

- *Multi-perspective:* means that the overall picture of the situation is constructed from elements recognised in various perspectives of the battlefield. For example, an Attack is characterised by a number of perspectives such as type, thrust, target, deployment (see Figure 9).
- *Multi-membership:* refers to the fact that within each perspective, several alternatives may coexist simultaneously. For example, an Attack may use both tanks and air forces.
- *Hierarchical:* means that low-level indications are used as building blocks for higher level indications.

The situation recognition process is mostly directed bottom-up. Occasionally, however, correlations between indications at the same level may provide horizontal evidence as well. For example, recognising the tactics of an attack may suggest evidence regarding the target of the attach and vice-versa (indicated by horizontal links in figure 9).

The authors divide the situation assessment process into a cycle of steps:

1. *Initial Findings Accumulation.* This starts with the presentation of an initial set of specific facts about the situation.
2. *Hypothesis Generation and Evaluation.* The recently obtained findings are integrated into the existing evidence, and trigger a chain of deductions pointing at alternative hypotheses. An attempt is made to see if an overall explanation is clear and if a global interpretation may be drawn.

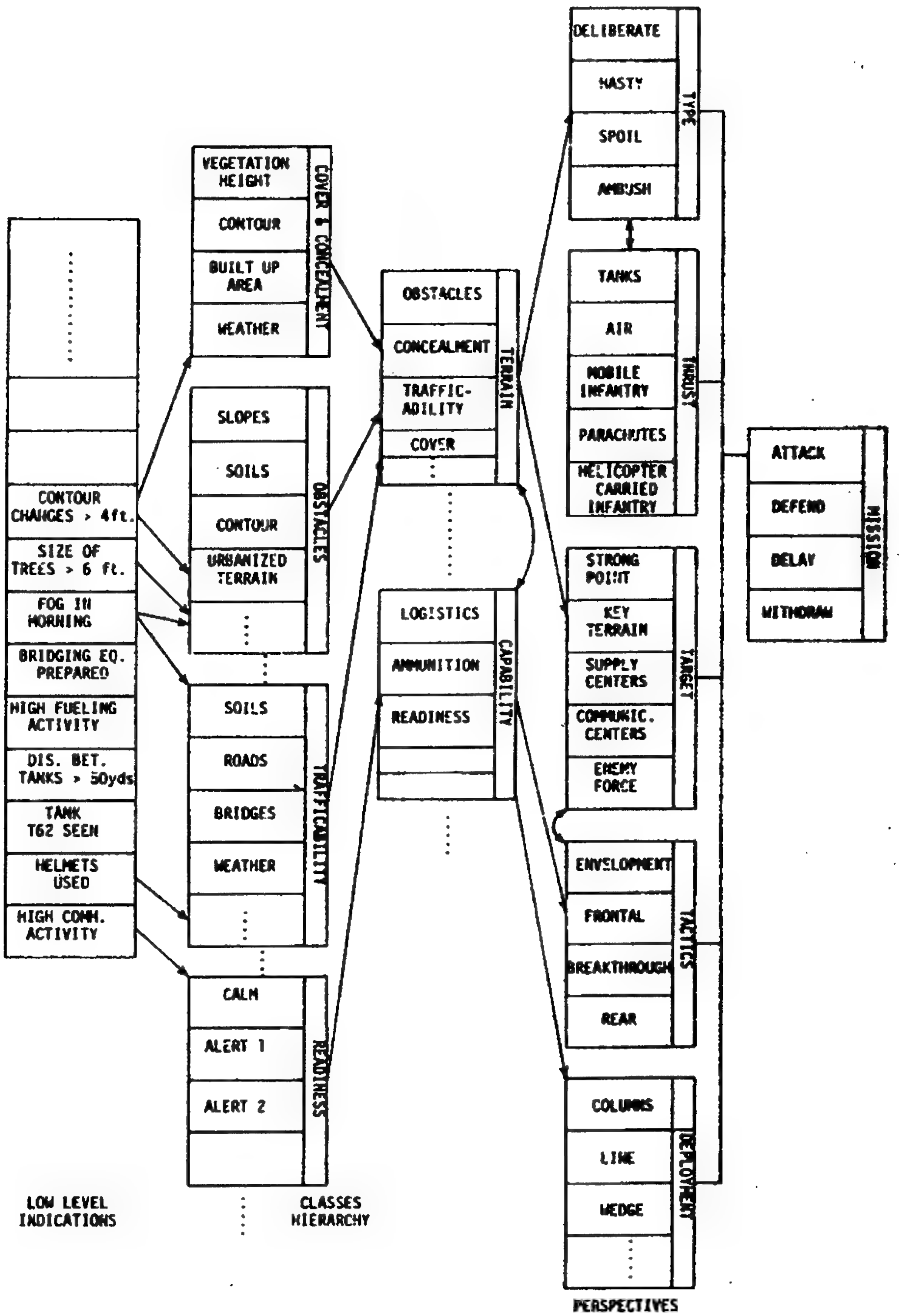


Figure 9: Multi-perspective, multi-membership hierarchical structure of situation assessment task [4]

3. *Goal Setting.* In the case of many hypotheses, not all may be explored simultaneously. Often, new information may be required to differentiate hypotheses. Thus, goals need to be set upon which attention will be focussed in the near future.

4. *Information Sources Evaluation and Selection.* Information sources which may be relevant to the current goal need to be identified and evaluated.

5. *Findings Sorting by Goals and Hypotheses.* As new findings are accumulated, they should be organised and prioritised with respect to existing knowledge. Findings should not be ignored just because they do not contribute to the current gals or the previously activated hypotheses.

6. *Evidence Integration.* Once the relevancy of the new findings are identified, they are integrated with the existing findings, not just added to them.

7. *Termination.* The situation assessment cycle is interrupted or terminated whenever one of the following conditions hold. Otherwise, the cycle returns to stage 2.

   - A decision may be reached with regard to the true situation in all aspects of the environment. All findings are explained by this interpretation, and no hypotheses justify further exploration.
   - Several active hypotheses have not yet been settled; however, the cost of removing the remaining uncertainty is relatively high compared to the expected information gain and the impact on the overall plan.
   - New developments force the decision-maker to terminate information acquisition, and assess the situation as best as possible with the existing evidence.

8. *Summary Composition.* An summary of findings is produced.

The authors describe the design of an intelligent computer aided situation military assessment system as consisting of three major tasks.

1. *Elicitation and computer representation of the necessary military knowledge for battlefield assessment.* This includes, for example, the characteristics of the various situations and the information sources that are available to the commander and their cost and reliability. The problem focuses on capturing the assessment process in a fashion that can later be used by inference algorithms for situation assessment.

2. *Modelling the reasoning and inference processes which take part during situation assessment.* These include recognising patterns of military indicators, evaluating and choosing information sources, and composing a picture of the battlefield.

3. *Design of a Human-Machine interface.* This includes the design and development of the language and other means by which a commander interacts with the system to make optimal use of its capabilities.

The authors note that:

> "It is not a simple task for a commander to verbalise and spell out the reasoning process which guided him in the analysis of a certain situation. On the other hand, a basic requirement for any intelligent computer system for military command is a systematic and structural representation of military knowledge. The transfer of knowledge from expert human beings to a computer system requires, therefore, two elements. The first is the development of an information structure to accommodate the expert's knowledge. The second is an elicitation technique by which the necessary military knowledge is extracted from expert commanders, manuals, and existing data bases. Of course, the information structure must be designed with the elicitation requirements in mind so that an optimal military knowledge base will emerge.
>
> The elicitation of military knowledge presents unique problems which stem from the fact that recent years have seen very few real large-scale battles. As a result, statistical battle data are not available, and the number of officers with actual battle experience is decreasing. This implies that a military knowledge base can rely only to a limited extent on previous experience. Rather, it will have to rely extensively on subjective and judgemental understanding of the overall doctrine of the opponent"

The authors describe the requirements for knowledge representation and elicitation to include the following:

1. *Compatibility with Human's Cognitive Processes.* The most fundamental requirement for any knowledge elicitation technique is the compatibility with the knowledge that a human expert can provide adequately. For example, humans are poor at statistical judgement so a system that requests probability estimates (eg. many rule-based expert systems) will not accurately capture the knowledge of the expert.

2. *Group Elicitation.* To avoid personal bias, mistakes, or lack of knowledge in a given individual, each component of the knowledge base must be produced by a team of experts.

3. *Modularity and Efficient Integration.* The building of a knowledge base cannot be completed by a single team. Thus, there needs to be a framework for integrating multiple, possibly conflicting, sources of knowledge.

4. *Elicitation from Existing Sources.* A great deal of the required knowledge base may exist explicitly or implicitly in textbooks, field manuals, and tactical operations databases. The elicitation techniques should make provisions for utilising these sources as much as possible.

5. *Minimal Burden on Experts.* The elicitation process is, by its nature, a lengthy process that requires significant intellectual efforts. Therefore, the more that can be done to facilitate the process, the higher the chances are to gain cooperation.

6. *Ease of Update.* The knowledge base will have to pass many iterations in which elements will be modified, deleted and others will be added.

7. *Computational Efficiency.* Efficient representation and storage is of great importance, not only for economic considerations, but due to the human factors of communication between machine and user.

3.1.2 Paradis et al, 1997

Paradis et al [36] present a generic model for situation and threat assessment influenced by human mental processing. This model results from research into decision support systems at the Defence Research Establishment Valcartier (DREV) in Canada. Key concepts in the model derive from the JDL model developed by a subpanel of the Joint Directors of U.S. Department of Defense Laboratories (JDL).

Data Fusion (DF) is defined as a hierarchical process to manage data and information gathered from a variety of sources that may be required by commanders for decision making. This includes sensor observations, topographic and environmental data, data describing capability and availability of targets, and information regarding doctrine and policy. Integrating this information into a sensible consistent picture involves dealing with conflicting reports, errors, deception, incompleteness and ambiguities about events or behaviours.

A complete data fusion system can be decomposed into four levels:

Level 1 Multi-source data fusion (MSDF)

Level 2 Situation Assessment (SA)

Level 3 Threat assessment (TA)

Level 4 Process Refinement through Resource Management (RM)

Each level deals with a higher level of data abstraction. Level 1 data fusion uses mostly numerical, statistical analysis, while levels 2,3, and 4 use mostly symbolic methods.

Multi-sensor data fusion (MSDF) is described as being concerned solely with individual objects. This includes estimating current and future positions for each hypothesised object, and inferring the identity or attributes of the objects.

Situation assessment (SA) develops a description of current relationships among objects and events in the context of the operational environment. This includes:

- Object Aggregation - establishment of relationships among objects including temporal relationships, geometrical proximity, communications links, and functional dependence.
- Event / Activity Aggregation - Establishment of relationships among diverse entities in time to identify meaningful events or activities.
- Contextual Interpretation / Fusion - Analysis of data with respect to the context of the evolving situation including weather, terrain, sea-state, or underwater conditions, enemy doctrine, and socio-political considerations.

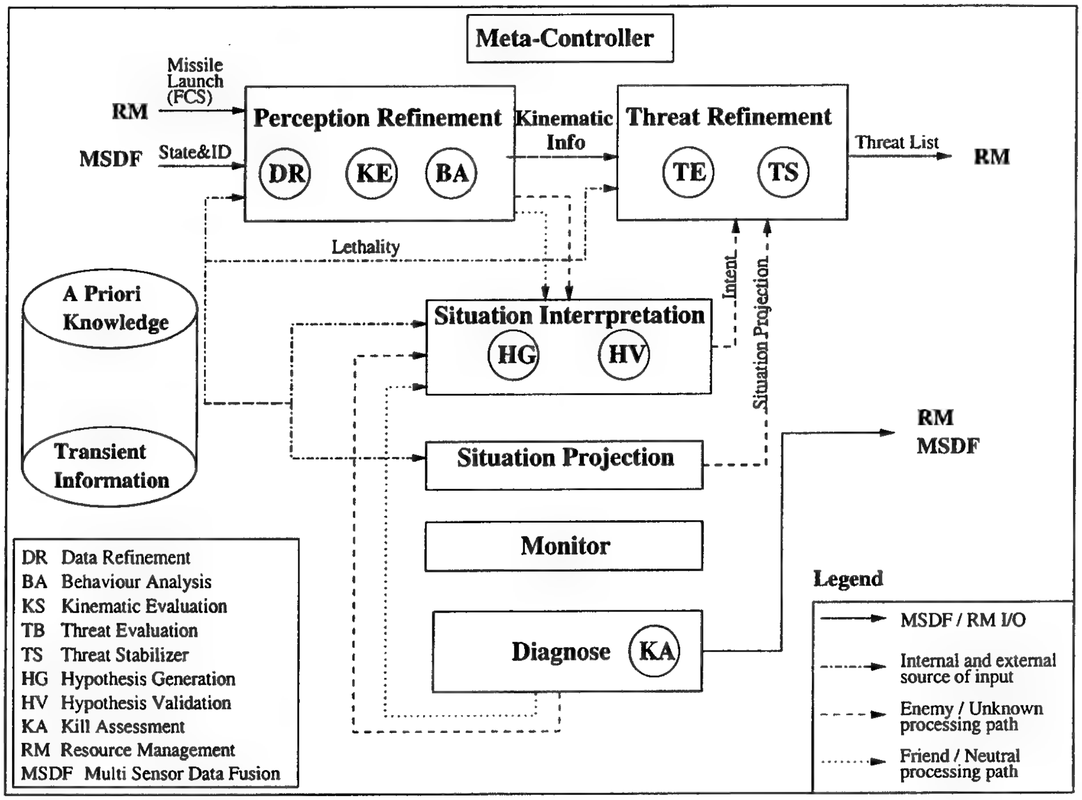


Figure 10: Generic model for Situation and Threat Assessment [36]

- Multi-perspective Assessment - Analysis of data with respect to three perspectives: (1) the blue (friendly) force; (2) the red (enemy) force; and (3) the white (neutral). It considers how the environment affects red and blue perspectives.

Threat Assessment (TA) focusses on the details necessary for decision makers to reach conclusions about how to position and commit friendly forces. By coupling the results of situation assessment with information in technical and doctrinal databases, TA develops and interprets a threat oriented perspective of the data to estimate enemy capabilities and lethality, identify threat opportunities, in terms of the ability of own forces to engage the enemy effectively, estimate enemy intent, and determine levels of risk and danger.

Resource Management (RM) examines and prioritises what is unknown in the context of the situation and threat and then develops options for collecting the required information by cueing the appropriate sensors and collection sources.

Figure 10 shows the general model of Situation and Threat Assessment (STA) described by [36]. It attempts to present the STA problem in the context of Naval Command and Control operations.

Here, we describe the parts that are relevant to situation assessment: perception refinement and situation interpretation.

The Perception Refinement module relates to low-level information such as entities (track data) and groups of entities (clusters). There are three subprocesses Data Refinement (DR), Kinematic Estimation (KE), and Behaviour Analysis (BA).

Data Refinement (DR) refines data by examining track attributes (position, identity) from MSDF for incompleteness and contradictions and attempts to establish relationships between these entities in order to form clusters. These relationships may be in terms of proximity, functionality, or dependency, and will be based on judgements using additional data sources.

Kinematic Estimation (KE) estimates kinematic parameters for weapon engagability calculations (closest point of approach, mean line of advance, time of flight). This is done by examining the history of each track.

Behaviour Analysis (BA) analyses entities or clusters in order to help refine the data set and provide the necessary cues for interpretation and understanding of the tactical situation. These cues are based on DR analysis (kinematics), from existing knowledge sources, and from transient information (electronic emissions, information from participating units). It includes functions such as corridor correlation and maneuver / pattern identification.

Situation Interpretation explains the presence of perceived entities and determines the intent of enemy or unknown tracks. This involves two sub-processes: Hypothesis Generation (HG) and Hypothesis Validation (HV). The HG process generates hypotheses about the probable situation that has caused the observed tracks. These hypotheses are obtained on the basis of the results of perception refinement (PR) and existing knowledge. The HV process checks these hypotheses for inconsistencies, conflicts of information, and possible inaccuracies due to incomplete data.

Goals for the implementation of the STA model are said to include:

- refinement of understanding of human cognitive processes for achieving situational awareness by conducting a top-down analysis of the problem. This will define the human's decision requirements according to established cognitive engineering methodologies.
- study to acquire expertise in the area of cognitive science, situation awareness, information warfare, and uncertainty management.

3.2 The SATE system

The Situation Assessment and Threat Evaluation System (SATE) [30] was developed by the U.K. Admiralty Research Establishment (ARE). Miles et al [30] describe situation assessment as "all those tactical assessments which are required to provide the command with a concise statement of the current situation and to support decisions on the allocation of resources." Input to the process is based on physical objects, their position, movement and identity. A number of assessments are described, including:

- *General Situation.* The disposition of own forces and enemy forces is the foundation of situation assessment on which more specific assessments can be based.

- *Threats.* Asseessment of both direct and indirect threats would be a useful function because a large amount of knowledge is required and feasibility calculations have to be performed in a short time-scale.
- *Mission Assessment.* This describes the current position against the objectives of the mission (for example, to maintain a free passage for shipping, to arrive at a destination at a given time, to escort other units).
- *Outcome of Actions.* This is of great importance to a commander because it may determine the next course of action. This must be provided rapidly during battle to be useful (eg. to conserve ordnance).
- *Weapon System Geometries.* This considers possible conflicts arising from the positioning of weapon systems in relation to one another.
- *Rules of Engagement.* These are often important in determining when and how to respond to a threat.
- *Plan Monitoring.* This monitors constantly whether plans are being followed, and alerts the command to any discrepancies.
- *Surveillance.* This attempts to estimate "the extent of our knowledge of the enemy forces and the enemy's knowledge of our forces."

The fundamental problem facing situation assessment is that complete, high-level interpretations are required, and usually only incomplete, low-level information is available. Three approaches are suggested to overcome this problem.

- *Group Formation.* By grouping data to reveal relationships, it may be possible to explain the behaviour of elements within an area of interest. This requires a hierarchical data structure to represent grouping of elements (vehicles) and a means to specify the construction and membership criteria of the groups. This must be represented in the form of rules. The following types of groups are identified:
  - *Functional Groups.* These consist of similar types of vehicles performing the same function. Some types of vehicles (eg. ships) can carry out several functions and will thus be members of two or more groups.
  - *Interacting Groups.* These are composed of two or more functional groups that can be seen to be a part of a common objective.
  - *Allegiance Groups.* In some circumstances, more than one interacting group may be required to explain a situation.
- *Plan Recognition.* In general, vehicle data will be incomplete, and attempts to form all functional and interacting groups will fail. This incompleteness can be overcome by considering a goal-driven approach. Goals in an interpretation can be described as "expected patterns." Given descriptions of such patterns, it may be possible to match the partial data against plans to infer details that cannot be directly perceived. In the military sense, these behaviour patterns are referred to as tactics. In AI terms, they are considered as "plans" or "scripts".

- *Prediction.* It may be possible to predict the future development of a situation, allowing "what-if" facilities to be available to commanders. Two main approaches to this are:
  - *Stored Plans.* If a plan is recognised at an early stage, its structure can be used to predict future events.
  - *Simulation.* Given a tactical picture, future events could be predicted by playing through actions from both sides starting from the current situation.

It runs on a VAX 11/780 running VMS using the MXA[1] expert system tool. SATE is built on top of an existing Data Fusion system.

The data-fusion system has three levels of knowledge: tracks, multi-tracks and vehicles. The SATE system adds functional groups, interacting groups, and allegiance groups. The inputs to SATE are a set of vehicle hypotheses. Interactions between different groups (such as a reconnaissance aircraft providing targeting data for an anti-surface ship group) are represented by hypotheses at the next level. Finally, groups of vehicles with the same hostility are created. A primary output of the system is a list of threats.

SATE contains "about twenty" knowledge sources, each with an average of three large and complex rules. These can be divided as follows:

1. Assessment of possible functions of vehicles (and thus groupings)
2. Assessment of possible interactions between function groups
3. Determination of threats (including prioritisation)
4. Determination of options for action
5. Assessment of enemy knowledge

The knowledge acquisition process was as follows. Two thirty-minute scenarios were developed, combining principal constituents of task group configurations and their defences with typical attach conditions. For each scenario, time-lines of significant events were specified, such as the appearance of a new hostile platform, or the receipt of further information on an existing platform. An expert and a knowledge engineer built up descriptions of the situation at each event, providing indications of the information required, the strategy adopted, and the initial high-level rules.

3.3 The SAP system

The Situation Assessment Prototype (SAP) [9] was developed by the U.K. Defence Research Establishment Maritime Division. Although not stated, it may be a successor to the SATE system (see previous section).

The main function of SAP is to generate a "situation display" and a "threat lies". To do this, it makes the following aseessments:

[1]Multiple Expert Architecture

- *Vehicle Assessment.* These are made on the basis of information supplied by a data fusion module. Knowing the type of each vehicle, it assesses the capability and role (assuming that particular types of vehicles perform certain roles.)

- *Vehicle Grouping.* This involves forming a single representation (hypothesis) to represent two or more vehicles that are deemed to be acting together. This is done by forming tentative groupings between a new vehicle and all groups with compatible identity, type and position. Tentative groupings are reviewed periodically until a single tentative relationship remains.

- *Vehicle Relationships.* Simple relationships between vehicles are postulated by SAP. For example, a Targeting Relationship involves a vehicle supplying targeting information to other vehicles or groups of vehicles.

- *Threat Assessment.* Vehicles and groups are assessed as threats by examining their "category" (capability?), time to reach detection range, time to reach engagement range, and time to deploy a weapon against own ship.

- *Threat Prioritisation.* Threats are rated by category and time to deploy a weapon against own ship.

SAP employs a blackboard knowledge-base model. A special purpose knowledge-base shell was developed in Ada. In SAP, knowledge is captured as a set of simple rules. A common template for all rules is employed. This includes:

1. Summary - a plain English version of the rule.

2. Rule Specification

   ```
   RULE SXXXnnn <Name>
     if <condition 1>
     and <condition 2>
     ...
     actions
   END RULE SXXXnnn
   ```

3. Assumptions / limitations - For example, assumptions and limitations concerning the data type

4. Definition of Criteria - explanation of conditions or reference to detail elsewhere

5. Supporting Operations - to avoid large complex rules, parts can be separated as supporting operations

6. Hypotheses - a list of hypothesis types

7. Data Structures - any blackboard data structures used

Knowledge elicitation for SAP was performed with an experienced ex-navy officer. Structured interviews based on scenarios were used to identify key concepts in situation assessment. Mock-up scenarios were generated using an object-oriented battle modelling toolkit, and these were used to refine elements of the knowledge base.

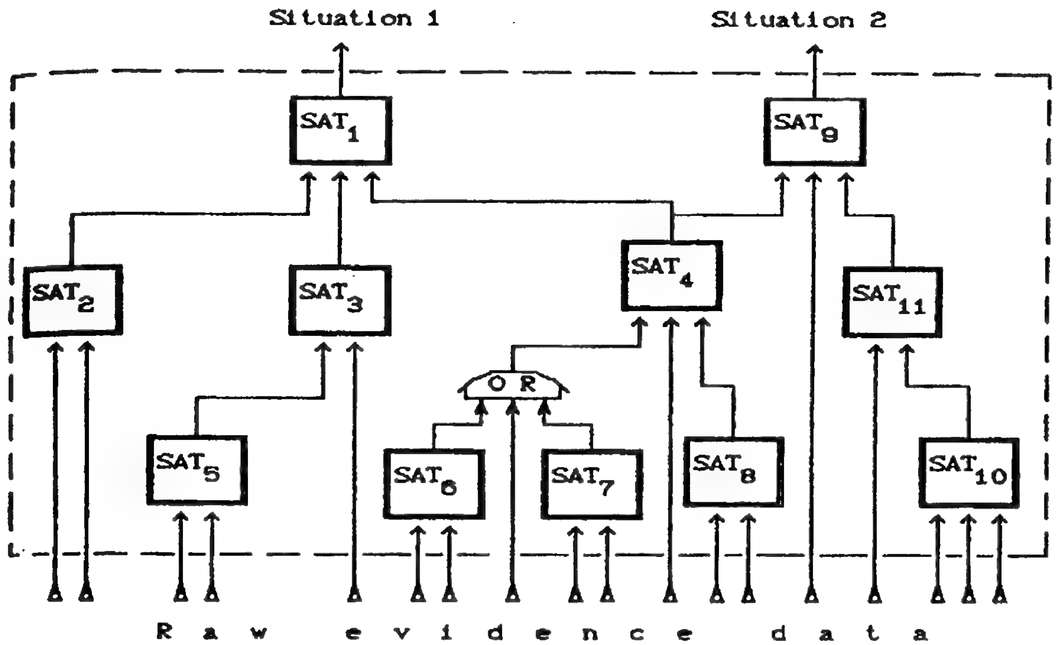


Figure 11: Hierarchy of tasks in situation assessment problem [21]

3.4 Stochastic Temporal Models (Kirillov, 1994)

Kirillov [21] describes a situation assessment system using a "hierarchical matched filtering paradigm". The described approach supposes that situation assessment tasks can be split into a hierarchy of subtasks ie. the outputs of some subtasks are allowed to be used as the inputs of others (see Figure 11). Kirillov points out that this implies a syntactical uniformity of the outputs and inputs of sub-tasks. Each task may be considered a filter whose output is a function of its inputs, and of spatial and temporal attributes.

A good filter should be reversible in the sense that for a given output message, it should be possible to determine the corresponding input conditions.

Kirillov presents a "Calculus of Judgements about Events", which considers the processing of evidence, default reasoning, spatial and temporal representations. The requirements for the temporal representation are stated as:

1. temporal knowledge used to specify tasks should be in a declarative rather than procedural form.
2. messages concerning detected events should contain references to the time axis.
3. both time instants and intervals should be treated. Imprecision of both events and knowledge should be managed.
4. both forward and backward modes of reasoning should be supported.

To handle imprecision, the calculus is based on a multi-valued logic. Kirillov states that Allen's temporal representation system fails to meet the requirements necessary to represent the domain of knowledge provided by human experts. This is, in part, because it is impossible to make explicit references on the temporal axis for the times at which events occur (ie. there are no metrics). Instead, a stochastic time relation "Forecast" is proposed for both time instants and intervals. For a given instant t_1, another forecast imprecise instant t_2 may be calculated so that t_2 is to the right of t_1 at a random distance, the mean and variance of which are specified. Rules are given for reasoning with Forecast relations.

Kirillov [21] describes a prototype situation assessment system developed between 1986-1988 for IBM/370 mainframes. The core of the system is an expert system shell for building situation assessment decision support systems for military intelligence applications.

3.5 Situation Assessment using Belief Nets

Belief nets (see Section 8) are directed graphs that represent the relationships between random variables. In situation assessment, they can be used to define the relationships between evidence and hypotheses. Macfazdean and Barber [28] describe an application of belief nets to naval situation assessment. A belief net is used to predict the intent of a vehicle based on observations of its behaviour.

A number of symbolic evidence variables are defined (see Table 3). These correspond to simple interpretations of low-level data. For example, high speed might be assigned to any vehicle moving faster than 300 m/s. A number of hypotheses are generated (see Table 4). Top level hypotheses (Top) are defined in terms of both intermediate hypotheses (Int) and low-level evidence variables. However, evidence variables that determine intermediate hypotheses are not also used directly for top-level hypotheses. The relationships are expressed in terms of conditional probabilities. The authors state:

> "Obviously, judgement and knowledge of the operating environment are necessary to construct such data, which is one of the criticisms of the Bayesian approach to probabilistic reasoning."

The described system was built using the IDEAL belief net package. Performance of the net was evaluated using a simulation environment. A scenario generator models target motion and sensor measurements. Data is fed into an evidence processor which maps sensory data to symbolic evidence. This is interfaced to the belief net via an evidence interface. By running various scenarios, it is possible to evaluate changes in the situation assessment hypotheses over time.

3.6 Bluff in Situation Assessment

Bluff is false information deliberately introduced into a person knowledge base. In a military context, commanders have to deal with bluff, either by interpreting the enemy's

Name	Values	Description
Altitute	low, med	Vehicle altitude characterisation
Speed	med, high	Vehicle speed characterisation
IFF code	iff3, iff4	iff4 normally occurs for friends, iff3 possibly occurs for any manned aircraft
bearing sector	red, yellow	Characterises direction of vehicle relative to ship, from which some inference of intent may be made
land	true, false	A piece of land (island, rocks, etc) is in the sensor resolution cell
closest point of approach (CPA)	cpa1, cpa2	CPA values represent an approach relatively near ship, from which the vehicle could easily maneuver to a collision course (cpa1), and a larger CPA from which a maneuver to hit the ship would be more difficult
range	red, yellow	Characterise range in terms of closeness to ship
altitude rate	alt-, alt+	Indicates whether altitude is increasing or decreasing
popup	true, false	A detection occurs at relatively short range
emitter character	etrck, esrch	A radar emitter is either tracking (spotlighting) the ship, or is searching

Table 3: Symbolic Evidence Variables for Belief Net [28]

Name	Type	Description
MSL	Int	Vehicle is a missile
HOS	Int	Vehicle is hostile
FRD	Int	Vehicle is a friend
COM	Int	Vehicle is a commercial aircraft
FA	Int	False alarm. No vehicle present
ATTACK	Top	Intent of vehicle is to attack
BENIGN	Top	Intent of vehicle is benign

Table 4: Belief Net Hypotheses [28]

misleading activities and communications, or by themselves generating false information to bluff the enemy. In World War II, bluff was used extensively via the communication of misleading information.

Lelouche and Doublait propose a model for representing an actor's bluff intentions and beliefs [27]. The model provides a logical formalism for representing bluff and belief, but provides no inference rules for manipulating such knowledge. The knowledge representation is designed to allow modelling of an automatic player in a deterministic game named "Sigma-File", which is primarily a game of bluff. It is suggested that the representation could be extended for more complex environments.

4 Representation of Submarine Tactics

4.1 Situation Assessment in Anti-submarine warfare

There has been a significant interest in decision problems related to Anti-submarine warfare (ASW). Some of these studies have already been described (see Section 2.4). Studies of human performance in ASW situation assessment are described here. These studies [54, 55] attempt to model the decision processes in order to understand the uncertainty and possible errors that arise in different situations. The modelling effort includes several parts:

- Modelling an optimal decision-maker.
- Identifying factors relevant to human performance.
- Constraining the model according to known human limitations.
- Comparing the model with human behaviour found by experiment.

The task facing an ASW commander includes the problem of identifying the position and intent of hostile agents from passive sonar contacts. There will be a number of sonar sensors distributed amongst the members of a naval battle group. Thermal gradients in the ocean focus underwater sound in convergence zones, which limit the areas where passive sonar can detect a target. In an m-convergence zone environment, a single bearing-only sonar measurement can be viewed as $m + 1$ feasible mutually exclusive bearing / range measurements. Given n sonar contacts, there exist $(m+2)^n - 1$ possible hypotheses about the origin of the n contacts.

Wohl et al [55] describe a normative model that transforms contact data from multiple distributed sensors into coherent state estimates of the submarines (positions and velocities). This includes:

- correlation of sensor measurements (contact data) with individual submarines (data association)

- updating of the state estimates of the submarines on the basis of the data associations, prior state estimates, and known dynamics of submarine motion (state estimation).

The contact data are assumed to be corrupted by zero mean random noise. A Kalman filter is used for state estimation. Data association prunes the hypotheses, or tracks, by associating measurements with an appropriate subset of the set of existing tracks.

In addition to the normative model, three descriptive factors were added to represent human biases, limitations and heuristics identified both from the cognitive literature (see 2.1) and prior experiments. These were:

- *The small number of concurrent hypotheses that Subjects appear to consider at any time.* This is a function of the limitations of human memory and attention.
- *The subjects' tendency to perceive their environment as less certain and fuzzier than it is in reality.*
- *The subject's tendency to overemphasise the value of the information provided by the sensors.* Subjects tend to overlook the uncertainty associated with measurements, and thus exaggerate the weight given to those measurements compared to the weight given to previous state information. This effect is known as "recency".

A sensitivity analysis compared the behaviour of the model with various values for important parameters to the behaviour of trained naval personnel. This indicated that the subjects tended, on average, to associate all tracks within three standard deviations of a measurement, reducing significantly the number of hypotheses generated. The subjects estimates also showed a strong recency effect, implying that they tend to overestimate the maneuverability of enemy submarines, and underestimate the error characteristics of the sensor input data.

4.2 Gonzalez and Ahlers, 1994

Gonzalez and Ahlers [13] describe an extension of Schank's script concept which can be applied to military tactics. This is used to control the behaviour of a submarine within a simulation. The system is implemented in CLIPS and was designed in collaboration with the U.S. Naval Training Systems Center (NTSC).

The authors note that:

> A script can be used in this application ... to express the set of steps (at either a high or low level) that are necessary to carry out the action required by the present situation. Within the context of a mission, there is a limited number of things that are generally expected in terms of actions to carry out and the expectations in regards to the possible situations.

In their representation, a *context* is composed of a set of rules and procedures which implement some action and detect transitions to other contexts when required. The rules form the basis for situation awareness. A context is composed of *Acts* (intermediate-level contexts that define maneuvers, situations or tactics relevant to that mission). Acts can be hierarchically decomposed into *Sub-Acts* (lower-level tactics or maneuvers). At any one time, there is a single "General Context" that defines the overall mission to be undertaken. Each context includes:

- Message handlers that initialise the appropriate objects in the simulation at the time of initial activation of the context of which they are part.
- Message handlers that execute certain actions during the time which the context is active.
- Rules which are applicable only when the context to which they belong is active. These rules assess the situation, prescribe actions to be taken within the context, or determine when a transition to another context is required.

The use of contexts partitions a large space of rules into clusters which are applicable in certain situations. Situation awareness is done through pattern-matching rules. Rules have a pattern in their premises that indicates the active context to which they are applicable. The use of context-dependent rules significantly reduces the overall search space.

Three types of rules are involved in decision-making:

- *Sentinel rules:* continually monitor the simulation data in order to recognise factors that can lead to a change in situation. They infer the situation (and thus the resulting context) from raw data.
- *Transition rules:* react to the situation identified by sentinel rules to determine which context should be activated in response to the situation.
- *Internal rules:* are used to make decisions in the current context. These rules do not cause transitions to a new context (for example, a baffle clearing act).

The "context" in the described approach appears to represent the overall goal or intention of the simulated submarines. An obvious question would be whether a realistic behaviour can be obtained by the use of mutually-exclusive contexts (ie. whether several goals might co-exist). The described system consists of 34 rules and 18 message handlers. The authors state that:

> While behaviour certainly cannot be considered to be as expert as an experienced submarine captain, it does provide a robust simulation of intelligent behaviour to a student who is in a simulated tactical encounter with it. These are in fact the objectives of this project.

4.3 Situation Spaces (Schmidt, et al 1989)

Schmidt et al [50] describe a planning method based on “situation spaces”. The approach is motivated by several factors.

They note that standard AI models for planning have traditionally employed a *predictability assumption*; that is, it is assumed that the planners model of the world as well as the effect of its actions on the world is complete and correct. This assumption is violated if an action fails to have its intended effect, or if relevant effects cannot be predicted by the planner. Even assuming predictability, there may not be a sequence of primitive actions to achieve a desired goal state. Classical planning models simply fail in this situation (ie. they provide no plan). The *effectiveness condition* specifies that planning must be successful. *Reactive planning* is described as the process of planning where either the predictability or effectiveness condition is violated.

In the ASW domain, the planner's goals are defined in relation to usually uncertain knowledge about the knowledge, goals, and actions of other hostile agents. The size of this search space is too large to be usefully pursued, even using reactive planners and heuristics. The “situation space” method of reactive planning is proposed to deal with militaristic “tactical spaces”.

The authors define a simplified ASW problem in which a single agent plans with respect to a single opponent. Each agent's goal is to destroy the other. There are three basic goals. *Preservation Goals:* include avoiding detection by other agents, avoiding collision with other agents, and avoiding attack and destruction by other agents. *Information Goals:* include knowing the present and type of ship of other agents, their location, speed and direction. The overall goal is to destroy the other agent through attack. Each agent has three basic actions available: setting speed and direction of own ship, and firing weapons.

Several important observations about the problem are made. First, each goal is defined relative to the target agents. Second, the goals are not independent. Preservation goals are necessary to the achievement of information goals as well as the overall goal. Third, the information goals must be achieved in order for the overall goal to be achieved. Finally, the planner's goal is not necessarily defined by a single fixed conjunction of these goals. Rather, differing conjunctive combinations of these goals may be pursued by the planner at particular points in the problem.

A situation space is defined as a collection of situations each of which includes:

- A situation name
- A world state, which expresses the information that must be obtained from the present world state for planning to ensue.
- A goal expression, which describes the appropriate planning goal for the situation.
- Characterisation, which is an indicator of the relative advantage of the situation.
- Transitions, which are a collection of ordered pairs that map an expression (defining the conditions for the transition) to a situation name (the resulting situation).

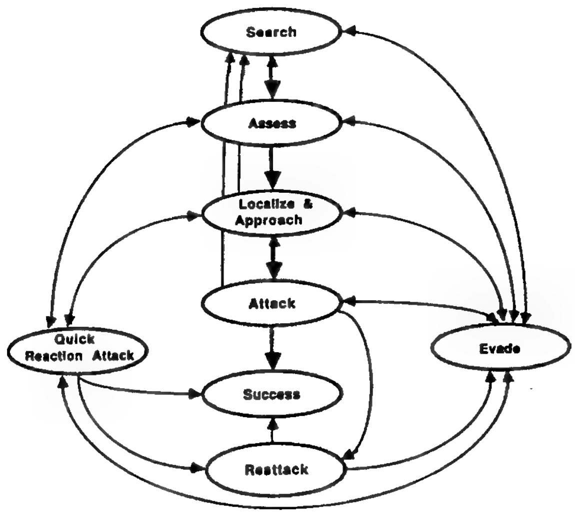


Figure 12: Situation space for a simplified tactical problem [50]

As stated, a situation space is similar to a set of contexts upon which scripts might be based in the approach of section 4.2. Note however, that the situations are defined declaratively so that a planner can deduce *how* to get into other situations as well as *what* actions to take within each situation.

The authors state that:

> "Since the overall plan is broken in to subplans, each plan is projected over a limited temporal (and indirectly spatial) span which serves in this problem domain to drastically limit the uncertainty about the actions of the other actor during the course of the subplan. ... This creation of subplans allows planning to proceed over spans where the predictability assumption may often be viable"

Figure 12 shows a situation space for a simplified tactical problem.

4.4 The SIAM system

The Submarine Interactive Attack Model (SIAM) [24] was developed by the U.K. Admiralty Research Establishment. By modelling the behaviour of submarines, it attempts to study strategies for engagements between two opposing submarines.

The SIAM model represents:

- platform characteristics (eg. hull, propulsion, sensors)
- data processing / data flows
- tactical decision-making

- weapons
- characteristics of the environment within which the two submarines interact.

For each submarine's suite of sensors, there are a series of *appraisal events*; a separate set exists for each submarine. The appraisal event represents an instant in time at which the tactical situation is examined from one submarine's point of view. The rates at which appraisal events occur is dependent on sensor characteristics and submarine motion. Tactical decisions result in a sequence of activities (eg. course changes), each of which is initiated by an *activity event*.

The authors state that "in scenarios of interest, the number of possible tactical situations that can develop is unbounded" and that it is "important that both infrequent as well as commonly occurring situations are adequately and economically represented in a meaningful way." In SIAM, each operating state is considered to be a well-defined phase of a mission (eg. the search phase). Within a state, a number of activities will take place, each one being initiated by an activity event. The activities themselves for a sequence of tactical instructions (eg. alter course to port by 30 degrees and increase speed to 15 knots, then dive to 200 feet). The complete set of sequences of tactical instructions for each operating state corresponds to "a compact body of knowledge about submarine operating philosophy."

Tactical decision making is described as the process of applying a set of rules that define what course of action will be taken in a perceived tactical situation. This is determined using rules (IF X THEN Y). For example: IF sonar contact is held THEN alter course towards the target. It is claimed that SIAM provides a "substantial flexibility in the choice of tactics", but acknowledge that the tactical input is "pre-determined". No explanation is given of how the tactical rules are elicited.

4.5 The ANITA system

The ANITA system [15, 41, 40] has some impressive goals. ANITA aims to:

> "identify the strategy of adversaries (ASW surface ships or submarines) on the basis of observations (sonar, radar, periscope ...) concerning their kinematics and the use of their sensors, extra means and weapons. This identification process is based on an original model of abstract behaviour schemes. Artificial Intelligence techniques are used to achieve this function, such as temporal reasoning, hypothetical reasoning, and incomplete / uncertain data reasoning."

Unfortunately, few details of the system are available in the published literature. The authors state that the knowledge representation model includes:

- The *mission and partial objectives* level, which is built after listing and analysing potential operations missions and general constraints (referring to politics, strategy, history, etc). Each mission is expanded in a coloured Petri net of partial objectives. These are operational tasks facing a particular target. Interconnections between

these objectives encode knowledge about temporal and other constraints between the tasks.

- The *actions* level, which contains possible procedures and elementary actions to (eg. maneuvers) to fulfil each local objective of each mission.
- The *techniques* level, which describes how actions are instantiated in "real life".
- *Acts*, which are the external manifestation of techniques. These describe the effect that the actor has on its environment.

Published details concentrate on low-level techniques, such as fuzzy inference rules and pattern matching. In an autonomous underwater vehicle (AUV) application, it is indicated that the petri-net transitions are based on high-level "events" such as "target classification", "object detection", or "failure detection". Development of the system has been done using an environment called X-IA, which provides a rule-based extension of traditional procedural languages (perhaps this is similar to CLIPS). Extensions to the language were developed to encode:

- Hypothetical reasoning techniques for real time execution constraints.
- Temporal reasoning techniques for the scheduling and optimising of actions.
- A distributed architecture to improve execution time.

4.6 The Maneuver Decision Aid

The Maneuver Decision Aid (MDA) [5] is a knowledge based expert system designed to simulate "the command level human maneuver generation process." The system was developed by the U.S. Naval Undersea Warfare Center. It aims to generate a theoretically optimal next maneuver recommendation to support the submarine commander's decision making process.

The maneuver generation process consists of three major parts: Situation Assessment, Goal Formulation and Constraint Satisfaction. Each identified goal in a situation leads to a set of constraints on the four motion variable (course, speed, depth, time), making it possible to consider the planning problem as a constraint satisfaction problem. In the submarine tactical domain:

> "it is often the case that no solution, or maneuver, exists that will satisfy all identified constraints. The typically overconstrained nature of the problem is countered by propagating the priorities of maneuver goals to produce *rated-constraints* and choosing a feasible solution from a maximal subset of the original constraints."

Within the MDA, maneuver goals based on a set of situation variables are posted on a blackboard by situation assessment knowledge sources. A maneuver goal may describe

objectives to be achieved or situations to be avoided. The goal generation process traverses down a hierarchy of goals. Each leaf goal is satisfied by a set of maneuvers. Each maneuver has specified *effects* on the four motion variables (course, speed, depth and time).

The available literature on the MDA ([5]) describes the constraint satisfaction process in some detail. Unfortunately, there are no details on how the situation assessment functions are performed.

5 Temporal Reasoning

5.1 Overview

Temporal reasoning involves reasoning about time. This is an essential part of many problems that involve planning, scheduling, and resource allocation. Planning a sequence of actions requires the use of knowledge about the interdependencies between events or intervals of time. These dependencies are represented as "constraints" or "assertions". For example "Obtain clearance from the control tower BEFORE entering the runway", or "the landing gear must be down DURING take-off and landing".

There are several important characteristics of temporal representations [1]:

- Temporal knowledge is often relative, and cannot be expressed in terms of absolute times or dates. A temporal representation must allow for *imprecise* knowledge.
- Often, exact relationships between two times are not known, but some constraints about the relationship may be known. A temporal representation should allow sufficient *uncertainty.*
- The scale of time may vary between problem domains. In some situations, it may be measured in days or years; others (eg. computer design) may deal with microseconds or nanoseconds.
- The model needs to support the notion of *persistence.* By default, the truth of a statement is assumed to persist, even though it may not be provable.

5.2 Description

A number of temporal representation systems are described by [1]. These include state-space approaches, date-line systems, before / after chaining, and formal models.

State space approaches provide a simple representation of change that is useful in some problem-solving tasks. A *state* is a description of the world (in terms of facts) at a point in time. Actions are modelled as functions that transform states (ie. cause facts to become true or false). By retaining a history of previous states a system can provide a simple model of time and change. Unfortunately, it is often too expensive (in terms of storage space) to maintain such histories.

Date-line representations involve tagging each fact by a *date*. Often, dates are represented by integers or real numbers allowing simple comparison of temporal ordering using numeric operations. However, in many applications precise temporal information is not available. Date-line systems cannot represent some forms of relative temporal information. For example, the fact that "A and B did not happen at the same time" cannot be represented. Using dates, either A is before B, or B is before A, or A and B are simultaneous.

Before / After chains represent temporal information directly using a graph. Temporal units are represented as nodes, and the relationships between them as edges. It is important to be able to propagate temporal information so that relationships can be deduced between times that are not directly related. This can be done at assertion time (by repeatedly adding new edges to represent deduced transitive relationships), or at query time (by deducing these relationships during a graph search). Section 5.3 describes the performance issues for these two approaches.

Formal models of time are systems (or calculi) in which temporal inferences are described in terms of formal logics (eg. [29])

Point representations define temporal relationships in terms of abstract, ordered points in time. Unlike date-line representations, points are not necessarily assigned a particular value on the temporal axis (such as an explicit time or date). Most point-based temporal representations allow intervals to be expressed using points as their boundaries.

Interval representations derive from the work of Allen [1], which is an extension of the Before/After chain approach. Here, the *interval* is the fundamental unit of time. Intervals have duration and may be related by a set of 13 possible relations (see Table 5). This allows the expression of certain relations that cannot be expressed with point representations. These include statements about the "disjointedness" of intervals. Allen describes an algorithm for propagating temporal constraints. A major problem with the algorithm is that it requires potentially $O(N^2)$ space to represent constraints between N intervals. To reduce this requirement, Allen introduces the notion of *reference intervals* which are used to group together clusters of intervals. Intervals within each cluster have their relationships fully computed. Reference intervals are treated as other intervals and form a hierarchy. Relationships between arbitrary intervals are deduced by searching for a path between their reference intervals, and propagating transitive relationships along the path.

Allen's interval framework has formed the basis for a large body of research in the field of temporal reasoning. Allen and Hayes [2, 3] describe an interval theory using a single relation MEET, which subsumes Allen's original theory. This theory includes a formalisation of the "beginnings" and "endings" of intervals (collectively known as "nests"), which act in many ways like "points", but can be distinguished from them. It also distinguishes *true intervals* which have at least two sub-intervals from *moments* which cannot be decomposed. Intuitively, moments are times during which an "instantaneous" event (such as a flash or bang) occurs.

Freska [10] describes a theory using semi-intervals (equivalent to nests). It is claimed that these are more "natural" entities from a computational and cognitive point of view. It also allows the representation of more "coarse" knowledge about relationships.

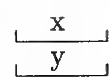
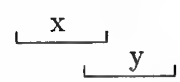
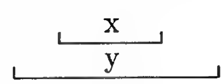
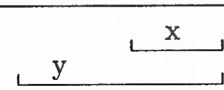
Relation	Symbol	Inverse	Meaning
x before y	<	>	
x equal y	=	=	
x meets y	m	mi	
x overlaps y	o	oi	
x during y	d	di	
x starts y	s	si	
x finishes y	f	fi	

Table 5: Thirteen possible relations in Allen's temporal representation (=, <, >, m, mi, o, oi, d, di, s, si, f, fi).

Allen's theory is based on *convex* intervals (ie. those without "gaps"). Ladkin [25, 26] investigates relationships between intervals which are *unions* of convex intervals (ie. they consist of convex sub-intervals with convex gaps between them). The resulting taxonomy of relations includes additional classifications (over the convex case) such as "intermingles with" and "surrounds". These can be modified by terms such as "mostly", "always", "partially", and "sometimes".

An important problem in the design of temporal reasoning systems is the complexity of the reasoning algorithms. Golumbic and Shamir [12] have shown that the problem of determining consistency for temporal constraint networks is NP-complete. However, many problems that arise in practice can often be solved quickly. Van Beek and Manchak [52] discuss ways of improving the efficiency of both path-consistency and backtracking algorithms. For path consistency, the bottleneck is the process of performing the composition operation[2]. Using heuristics to reduce the number of compositions can result in a factor of 10 improvement in speed. By combining a number of heuristics, the computation time to solve Benzer's matrix via path consistency was reduced from 137.7 seconds (Allen's algorithm) to 2.7 seconds[3].

5.3 Applications

A survey of temporal reasoning systems [56] compares the performance of six systems: TimeLogic, TimeGraph, MATS, Tachyon, TMM and TimeGraph-II. While they vary greatly in their underlying mechanisms and expressive capability, they can be broadly classified into two types:

- Those that use constraint satisfaction techniques at *assertion time* and have a fixed query time (TimeLogic, MATS, Tachyon)
- Those that build (incomplete) graph structures that require some search at *query time* to compute an answer (TimeGraph, TimeGraph-II, TMM).

The study examined the performance of the systems in a complex temporal reasoning task: planning a train timetable. Given a large number of constraints, the systems were required to determine a workable schedule. The study concluded that:

- For large temporal datasets, where data is added incrementally: systems based on incomplete graphs are the best option. The savings at assertion time are so substantial as to outweigh the additional costs of queries.
- For large datasets, where assertion time is not an issue, constraint satisfaction systems can be used. However, since they use fully connected graphs, the memory requirement will be large. In the trials, TimeLogic and MATS used up the available

[2] If the relationship between A and B is r, and the relationship between B and C is s, the *composition* $r \circ s$ is the relationship between A and C. Since both r and s have 2^{13} possible labels, the size of a lookup table would be 2^{26} entries, which is impractical.

[3] Timings are for a Sun 4/25 with 12 megabytes of memory. Benzer's matrix is defined in [52] and involves an interval-algebra network of 145 intervals that arose from a problem in molecular biology.

64 Megabytes of memory with only 2000 assertions, whereas other systems were tested up to 30000 assertions.

- For small datasets, all systems may be used effectively, and the choice will depend on the expressive features that are required.

The measured average query time shows that the penalty of non-constant query time algorithms is negligible in most cases.

The TMM ("Time Map Manager") system [8] offers facilities for reasoning about the persistence of facts, which are not provided by other systems. This type of reasoning is critical in many situations. In TMM, a *time map* is a graph in which vertices refer to *points* or *instances* in time. *Constraints* are directed edges that link points. Each edge is labelled with an upper and lower bound on the distance separating the points in time. An *interval* is a pair of points, such that the *begin* point precedes or coincides with the *end* point. A *time token* or *persistence* is an interval paired with a formula that represents an occurrence. Some facts are assumed to persist once they become true. Others (events) may have a known duration. TMM can resolve apparent contradictions by forcing the end of the earlier to precede the beginning of the later (persistences are designed so that their extents can be easily modified). TMM has been used in the FORBIN [31] planning system for mobile robots.

6 Spatial Reasoning

Spatial reasoning is the process of deducing relationships between objects in space. Applications for spatial reasoning include computer-aided design (CAD) and manufacturing (CAM) where it is important to be able to represent and formalise the relationships between objects in space. Geographic Information Systems (GIS) typically represent both spatial and non-spatial attributes of objects. Spatial and temporal reasoning are usually combined within planning systems.

As with temporal modelling, spatial data can be represented using precise information (such as points, lines, surfaces and geometric solids) or imprecise (or qualitative) information. A summary of qualitative spatial representations is given by Travers [51]. A problem with qualitative systems is that ambiguity tends to increase with composition of relations. This severely limits the deductive power of systems.

Several authors have studied similarities between space and time. Both temporal and qualitative spatial reasoning use classification hierarchies for relationships. These hierarchies form "lattice" structures, or partial orders. Thus, similar techniques are required to compute transitivity (or composition) of relations. Randell and Cohn [43] define a logic formalism for reasoning about space and time. They impose "continuity restrictions" upon the way in which relations can change over time. Thus, for example, a body A cannot change from being "wholly outside" B to "wholly inside" B without passing through a state of being "partially inside". These "transition networks" also form lattices.

Hartley [17] describes a uniform representation for time and space using conceptual graphs. A problem is described in terms of a graph, with both spatial and temporal constraints between "actors".

7 Coloured Petri Nets

7.1 Overview

Petri Nets are a graph-based representation for discrete systems. Like *finite state machines*, they describe states of a system and the rules that govern transitions between them. Unlike finite state machines, Petri Nets allow the simultaneous representation of multiple states using *tokens*. This makes them ideal for situations where it is necessary to model interacting processes. This includes systems that involve communication, synchronisation and resource sharing. Formal properties of Petri Nets allow them to be used in the specification, simulation and verification of systems. Typical applications are in the areas of communication protocols, distributed systems, control systems, work-flow analysis, and VLSI design.

7.2 Description

A Petri Net is a directed graph in which there are two basic types of nodes. *Places* are drawn as circles and represent states of the system. *Transitions* are drawn as rectangles and represent actions. *Input Arcs* connect places (called "Input Places") to transitions. *Output Arcs* connect transitions to places (called "Output Places"). The Petri Net can be *marked* by placing *tokens* inside places. When a transition has at least one token in each of its input places, it may be fired. When a transition fires, one token is removed from each of its input places, and one is added to each of its output places. In this way, tokens propagate throughout the Petri Net.

A Coloured Petri Net (CPN) is an extension of the basic Petri Net model that allows for different types ("colours") of tokens. In a CPN, each token carries a data value. *Arc Expressions* describe the values of tokens generated by transitions. Typically, input arcs are labelled by variables, the values of incoming tokens are matched to variables, and the output token is specified in terms of the input variables.

Given the design of a CPN, there are several different types of analyses that can be done. *Simulation* can be used to test a CPN model and determine statistics about the behaviour of the modelled system. By simulating various types of problems, it is possible to explore the robustness of the design. Important factors might include delays (how quickly a token traverses an output arc), failures (a token is lost), and errors (a token's value is corrupted). *Occurrence Graphs* (also called state spaces or reachability graphs) contain a node for each reachable state, and an arc for each possible transition from one state to another. Although state-spaces can be large, they can be constructed and analysed automatically. *Place Invariants* are equations which can be proved to be satisfied for all reachable states. This can be used to prove properties of the modelled system eg. the absence of deadlock.

The Actor model is a computational framework for building open distributed applications. Actors have an internal state that is modified in response to messages received from other Actors. The Actor model is a popular framework for simulation systems. An equivalence between Actors and CPNs has been shown [48], which can be used to formalise the semantics of Actor systems.

7.3 Applications

Design/CPN is a package developed by the University of AAarhus that allows the construction, editing, and simulation of large, modular CP-Nets with or without time delays. It also supports the analysis of state spaces, allowing the user to verify many different behavioural properties. A large amount of material related to CP-Nets, including the Design/CPN tool can be found at:

http://www.daimi.aau.dk/CPnets

CPNs have been used to model the European Train Control System (ETCS) [19]. The European train system involves many different railway companies which have mostly incompatible train control systems. The ETCS is a communication-based train control system which defines a standard uniform signalling system and user interface. This means that neither drivers nor locomotives need to be replaced when crossing borders. Speed supervision and brake intervention are performed by an on-board system that communicates via radio with track-side systems. This is used to maintain safety distances between trains, and to supervise operations such as the joining and splitting of trains. The ETCS model consists of two separate hierarchical models to represent to on-board and track-side systems. There are approximately 200 nets and 2500 transitions and places. It is being used to test the completeness of the specification, and to systematically derive test cases.

A naval Command and Control system is modeled with CPNs by [6]. The article describes a conceptual high-level naval defence model called SARA (Situation Assessment and Resource Allocation). The model consists of six interconnected processes. The environment is modelled by the *threat scenario*, which characterises the kinematic behaviour of a collection of air threats (anti-ship missiles emerging from hostile platforms). The system (ship unit) comprises the remaining processes: *sensor*, *weapon*, *track manager*, *action manager*, and *battle manager* responsible for target evaluation and weapon assignment (TEWA). Tokens model the internal states and communication between the processes. Model simulations are used to evaluate the survivability of the system under two weapon-assignment strategies: earliest intercept and random assignment. Results have proved compatible with simulations conducted using an object-oriented simulation environment. Advantages of the CPN approach to simulation over the OO approach are listed as: rapid prototyping, formal specification, system modularity, explicit representation of co-occurrence, easily manageable, formal analysis capability, shorter development cycle and maintenance. Weaknesses are: symbolic treatment limitations, granularity of modelling, lack of flexibility with numerical algorithms, and restricted capabilities of the design tool.

An Air-to-Air missile simulator is modelled with CPNs by [14]. IWS (Integrated Weapons Simulator) is an engagement simulator used by DSTO Australia to test algorithms for the guidance and control of Air-to-Air missiles. Since the algorithms may be large and complex, the system is designed to provide concurrent execution and remote execution of separate components of the simulation. It is thus important to verify that communication between components is correct. The user of the system controls the operation of the simulation and the behaviour of the target, and views the target from the missile's point of view via a GUI. The components of the system are: *GUI*, *target*, *radar*,

infrared, and *missile control*. By modelling the interaction between components, CPN simulations were used to visualise the flow of information within the system. Using occurrence graph analysis it was verified that the IWS simulation terminated correctly (with a "hit" or by the user). However, it was found that transitions could occur after termination which meant that the latest state may not be displayed. Thus, a new requirement of the GUI was identified: that it should force display updates after the simulation had stopped.

In time-dependent applications (eg. real-time systems), it is important to model timing constraints on interactions between Actors. In [34], CPNs are used to verify *schedulability* (ie. that messages are delivered by their deadlines) in Actor systems with timing constraints. Rather than using *threads*, concurrency is provided by a *control machine* which enables scheduling to be precisely controlled. In a sample application, the authors describe a software-controlled crane in terms of actors. The system (actors, control machine, scheduler, etc) is 1-1 mapped onto CPN-subnets. Using occurrence graph analysis, it can be verified (for example) that for each *Emergency* event, the crane's magnet is turned off within a specified deadline.

In [33], CPNs are used to model a nuclear waste management program. SADT (Structured Analysis and Design Technique) is used to provide work-flow descriptions of the functions to be performed by the program. The descriptions are represented using a number of CPNs corresponding to the different functions within the program. The CPNs are simulated to produce timed event charts that can be used for understanding the behaviour of the various functions under difference scenarios. They can also be linked together to validate the interaction and co-operation between different program functions.

The examples described here show how CPNs have been used to model the formal behaviour of several different types of systems. All are characterised by an internal structure of intercommunicating components. Further examples of industrial uses of CP-nets can be found at:

http://www.daimi.aau.dk/CPnets/intro/example_indu.html

7.4 Summary

Coloured Petri Nets (CPNs) are a graph-based formalism that can be used to describe the behaviour of interacting systems. Systems represented by CPNs can be simulated to explore their behaviour in response to various internal and external factors. Formal properties of Petri Nets allow certain properties of systems to be automatically verified. For example, it may be possible to show that only certain "correct" states can be reached (eg. avoidance of deadlocks).

8 Uncertainty Representations

8.1 Overview

It is important in representing knowledge to adequately deal with uncertainty. Uncertainty can arise from several sources [18]:

- *Reliability of Measurements.* All measurements have a degree of related uncertainty. It is important to represent the reliability of knowledge derived from measurements (eg. the reading is accurate to within plus or minus ten percent).

- *Probabilistic Information.* Certain types of information are inherently probabilistic. For example, "One in five hundred boys are affected by Pyloric Stenosis".

- *Inference Mechanisms.* Certain types of inferences used in expert systems are based on "default" or "common-sense" rules that may not actually apply in practice. In a "weak implication", premise and conclusion may be partially correlated. For example given "John is driving a Rolls Royce", we may infer that "John is rich". However, the conclusion is not certain since John may have borrowed the car, or John may be a chauffeur, etc.

- *Lexical Imprecision.* Knowledge is often represented using terms that do not have a precise meaning. Thus, uncertainty arises because of differences in *definitions.* For example, one expert may consider a *fever* to be a temperature of 37.5 degrees centigrade, whereas another may consider it to be 38.0 degrees centigrade.

Various numerical systems have been devised to represent and reason with uncertain information. These techniques are used extensively in *expert systems.* The main candidates are:

- *Bayesian calculus.* Uncertainty viewed as a probability, where probability can be interpreted as relative frequency, or degree of belief. Probabilities are numerical values in the range [0-1], where 1 means that an event always occurs, and 0 means that an event never occurs.

- *Dempster-Shafer theory.* Uncertainty is viewed as a degree of belief. These are numerical values in the range [0-1], where 1 means total belief, and 0 means lack of belief (which is different from *disbelief*).

- *Fuzzy Set theory.* Uncertainty is viewed as a degree of membership of a set. Set membership is expressed using a numerical value in the range[0-1]. The value 1 means that the object is a member; 0 means that it is not a member of the set, and intermediate values indicate *partial membership.*

- *MYCIN / EMYCIN.* The MYCIN representation was developed especially for knowledge-based systems. In MYCIN, uncertainty is viewed as a degree of confirmation. This is expressed numerically as a value in the range [-1 - 1], where 1 indicates that the evidence confirms the hypothesis, -1 indicates that the evidence "disconfirms" (refutes) the hypothesis, and 0 means that the confirming and disconfirming evidence is balanced.

In complex systems it is likely that a number of factors may account for an observation. Knowledge about the state of one variable may "explain away" other possible causes. The interdependency between evidence and belief can be expressed in graph diagrams. *Baysian belief networks* and *Markov networks* are two such formalisms. Network representations allow qualitative dependencies to be explicitly represented.

Bayesian networks are directed acyclic graphs in which each node represents a random variable, or uncertain quantity, which can take on two or more possible values. Arcs signify the existence of direct causal influence between the linked variables, and the strength of these influences are quantified by conditional probabilities [39]. Direct representation of causal influence minimises the number of relationships that need to be considered during inference procedures.

The issues of representing and reasoning with belief networks are complex and will not be further described here. For details, see [39, 20].

8.2 Applications

Most commercially available speech recognition systems use some form of inference network (eg. Markov models) to determine possible interpretations for spoken words.

The CLARET system [37] uses a network (or "rule-graph") representation to learn the interpretation of trajectories in space. An application of the system is the recognition of hand-written symbols. The system has also been applied to the modelling and recognition of aircraft maneuvers in the FSIM flight simulator [38].

Other applications (with references) of Bayesian networks are described by [20].

BOBLO is a system which helps in the verification of parentage for Jersey cattle through blood-type identification. Heredity is determined by genes which are placed in chromosomes. For the blood group determination of cattle, ten different independent blood-group systems are used. These systems control 52 different blood-group *factors* which can be measured in a laboratory. Heredity of blood type follows narmal genetic rules. For each blood group a Bayesian network represents inheritance of phenotypes. However, the model allows for possible errors in the registration of parentage. Another example the use of Bayesian networks in agriculture is a system that controls mildew in winter wheat.

Bayesian networks are used extensively in computer vision for the interpretation of images. These networks have also been used (at a "meta" level) to control the allocation of computer resources in the interpretation process.

The *VISTA* system developed by NASA filters and displays information on the propulsion system during space shuttle launches. Bayesian networks are also used to govern information retrieval processes within information systems.

A number of applications exist for medical diagnosis. The *Child* system helps in diagnosing congenital heart diseases. Whenever a "blue baby" is born in South-East England, the paediatrician calls a 24 hour telephone service at the Greater Ormond Street Hospital, and the clinician on duty determines a provisional diagnosis based on evidence provided. The clinician then decides whether to transfer the baby to the hospital. A Bayesian network helps the clinician verify a diagnosis. One belief network was built through dialog with heart disease specialists. Another network was built automatically using a "batch learning" process based on 151 cases with established disease. Tests showed that both models performed at a level similar to the hospital clinicians.

The *Pathfinder* system assists community pathologists with the diagnosis of lymph-node pathology. It consists of a Bayesian network covering over 60 mutually-exclusive diseases (the system does not diagnose multiple diseases). There are 130 information variables that can affect the diagnosis. The system recommends to the user which test to perform next. It has been shown to be as good as experts in its diagnostic accuracy.

9 Conclusion

This survey shows that Situation Assessment is a "Multi-perspective multi-membership hierachical pattern recognition problem"[4]. Faulty situation assessment will lead to poor decisions. For instance, 175 military aviation mishaps, and 88 percent of major aircraft accidents were attributed to poor situation asessment[16]. The domain experts know how to assess situations. To investigate the process we must elicit that know-how and be able to store it. There is no over encompassing technique for knowledge elicitation. It is suggested that knowledge representation and elicitation processes should [4]:

1. Be compatible with human cognitive processes;
2. Elicit from a group of experts at the same time where possible;
3. Efficiently integrate multiple expert views from different sessions;
4. Elicit from written sources where possible;
5. Minimise the burden on experts;
6. Use a knowledge representation that permits easy update, and;
7. Use computational effeciency to minimize communication between machine and user.

When people make decisions it is not on the basis of actual data, but on internal representation or perception of that data[57]. In addition, people structure their decision process with goals, which are specifications of desired situation. We "chunk" information, where complex patterns may be reduced to single symbol (or chunks) when existing knowledge is taken into account. Our decision-making process is governed by our limitations which include [57]:

1. Limited working memory;
2. Slow cognitive operation;
3. Retrieval of information is biased to information that is recent, frequently recalled, or relates to information currently active;
4. Slow and error prone when dealing with numerical operations; and
5. Poor projection in time and space;

This understanding of human decision making and knowledge representation techiques will guide our development of the Situation Description Language. This formal language is targeted at describing situations in the domain of submarine operations, including mission and tactical goals. The formal Situation Description Language will take into account the situational context and spatio-temporal relationships in the information.

10 Glossary

ASW Anti-submarine warfare.

AUV Autonomous underwater vehicle.

CDM Critical Decision Method. See 2.3.

DSS Decision Support System.

GOMS A cognitive task analysis language (Goals, Operators, Methods, Selection rules). See 2.4.

HCI Human-Computer Interaction.

ICAT Intelligent Computer-Aided Training.

PRS Procedural Reasoning System. See 2.5.

RPD Recognition-Primed Decision model. See 2.3.

WM Working Memory. See 2.1.

TARGET NASA's Task Analysis Rule GEneration Tool. See 2.7.

References

1. Allen, J. F. (1983) Maintaining knowledge about temporal intervals, *Communications of the ACM* **26**(11), 832–843.

2. Allen, J. F. & Hayes, P. J. (1985) A common-sense theory of time, *in Proceedings of the 9th International Joint Conference on Artificial Intelligence*, AAAI, pp. 528–531.

3. Allen, J. F. & Hayes, P. J. (1989) Moments and points in an interval-based temporal logic, *Computational Intelligence* **5**, 225–238.

4. Ben-Bassat, M. & Freedy, A. (1982) Knowledge requirements and management in expert decision support systems for (military) situation assessment, *IEEE Transactions on Systems, Man and Cybernetics* **12**(4), 479–490.

5. Benjamin, M., Viana, T., Corbett, K. & Silva, A. (1993) Satisfying multiple rated-constraints in a knowledge based decision aid, *in Proceedings of the Ninth Conference on Artificial Intelligence for Applications*, pp. 277–283.

6. Berger, J. & Lamontagne, L. (1993) A Colored Petri Net model for a naval command and control system, *in Proceedings of the 14th International Petri Net Conference, Chicago 1993, Lecture Notes in Computer Science, Volume 691*, Springer-Verlag, pp. 532–541.

7. Card, S. K., Moran, T. P. & Newell, A. (1983) *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale, NJ, USA.

8. Dean, T. L. & McDermott, D. V. (1987) Temporal data base management, *Artificial Intelligence* **32**, 1–55.

9. Edmonds, C. J., Ritchie, I. M., Lewis, S. A., Miles, J. A. H. & Narborough-Hall, C. S. (1992) Automated support for situation assessment, *in Proceedings of the Conference on Undersea Defence Technology*, pp. 300–307.

10. Freksa, C. (1992) Temporal reasoning based on semi-intervals, *Artificial Intelligence* **54**, 199–227.

11. Georgeff, M. P. & Lansky, A. L. (1986) Procedural knowledge, *Proceedings of the IEEE* **74**(10), 1383–1398.

12. Golumbic, M. C. & Shamir, R. (1993) Complexity and algorithms for reasoning about time: a graph-theoretic approach, *Journal of the Association for Computing Machinery* **40**, 1108–1133.

13. Gonzalez, A. J. & Ahlers, R. H. (1994) A novel paradigm for representing tactical knowledge in intelligent simulated opponents, *in Proceedings of the International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Gordon and Breach, Yverdon les Bains, Switzerland, pp. 515–523.

14. Gordon, S. & Billington, J. (1998) Applying Coloured Petri Nets and Design/CPN to an air-to-air missile simulator, *in* K. Jensen, ed., *Proceedings of the Workshop on Practical Use of Coloured Petri Nets and Design/CPN*, University of Aarhus, pp. 1–14.

15. Guidet, Y. & Pignon, J.-P. (1994) ANITA : A system reasoning on fuzzy knowledge (tactical decision support system), *in Moving Towards Expert Systems Globally in the 21st Century*, Cognizant Computer Corporation, Elmsford, NY, USA, pp. 1011–1015.

16. Hall, M. E., Maren, A. J. & Akita, D. (1997) Modeling situation assessment to improve pilot safety, *in Proceedings of the International Conference on Systems, Man and Cybernetics*, IEEE, New York, NY, USA, pp. 4163–4164.

17. Hartley, R. T. (1992) A uniform representation for time and space and their mutual constraints, *Computers and Mathematics with Applications* **23**(6–9), 441–457.

18. Henkind, S. J. & Harrison, M. C. (1988) An analysis of four uncertainty calculi, *Transactions on Systems, Man, and Cybernetics* **18**(5), 700 – 714.

19. Jansen, L., Meyer zu Hörste, M. & Schnieder, E. (1998) Technical issues in modelling the European Train Control System (ETCS) using Coloured Petri Nets and the Design/CPN tools, *in* K. Jensen, ed., *Proceedings of the Workshop on Practical Use of Coloured Petri Nets and Design/CPN*, University of Aarhus, pp. 103–115.

20. Jensen, F. V. (1996) *An introduction to Bayesian Networks*, UCL Press, London.

21. Kirillov, V. (1994) Constructive stochastic temporal reasoning in situation assessment, *IEEE Transactions on Systems, Man and Cybernetics* **24**(8), 1099–1113.

22. Klein, G. A. (1998) *Sources of Power*, MIT Press, Cambridge, MA, USA.

23. Klein, G. A., Calderwood, R. & Macgregor, D. (1989) Critical decision method for eliciting knowledge, *IEEE Transactions on Systems, Man and Cybernetics* **19**(3), 462–472.

24. Knapp, B. M., Dudley, A. R. & Ryder, J. S. (1987) Modelling techniques for simulation of submarine engagements, *Journal of the Operational Research Society* **38**(10), 891–898.

25. Ladkin, P. (1986) Primitives and units for time specification, *in Proceedings of AAAI-86*, AAAI, Morgan Kaufman, pp. 354–359.

26. Ladkin, P. (1986) Time representation: A taxonomy of interval relations, *in Proceedings of the 5th National Conference on Artificial Intelligence*, AAAI, Morgan Kaufman, pp. 360–366.

27. Lelouche, R. & Doublait, S. (1992) Qualitative reasoning with bluff and beliefs in a multi-actor environment, *in International Journal of Man-Machine Studies*, Vol. 36, pp. 149–165.

28. Macfazdean, R. & Barber, K. S. (1995) Simulation of multilayer belief nets for situation assessment, *in Proceedings of The Summer Computer Simulation Conference*, San Diego, CA, USA, pp. 483–488.

29. McDermott, D. (1982) A temporal logic for reasoning about processes and plans, *Cognitive Science* **6**, 101–155.

30. Miles, J. A. H., England, E. M., Faulkner, H. C. & Frampton, S. P. (1988) Knowledge based techniques for tactical situtation assessment, *in Proceedings of the MILCOMP '88 Conference: Military Computers, Graphics and Software*, pp. 313–318.

31. Miller, D., Firby, R. J. & Dean, T. (1985) Deadlines, travel time, and robot problem solving, *in Proceedings of the 9th International Joint Conference on Artificial Intelligence*, IJCAII and AAAI, IJCAII, pp. 1052–1054.

32. Minsky, M. (1975) A framework for representing knowledge, *in* P. H. Winston, ed., *The Psychology of Computer Vision*, McGraw Hill, New York, USA, pp. 211–277.

33. Mortensen, K. H. & Pinci, V. (1994) Modelling the work flow of a nuclear waste management program, *in* R. Valette, ed., *Proceedings of the 15th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science, Vol 815*, pp. 376 – 395.

34. Nigro, L. & Pupo, F. (1998) Using Design/CPN for the schedulability analysis of actor systems with timing constraints, *in* K. Jensen, ed., *Proceedings of the Workshop on Practical Use of Coloured Petri Nets and Design/CPN*, University of Aarhus, pp. 271–285.

35. Noble, D. F. (1989) Schema-based knowledge elicitation for planning and situation assessment aids, *Proceedings of the IEEE* **19**(3), 473–482.

36. Paradis, S., Chalmers, B. A., Carling, R. & Bergeron, P. (1997) Toward a generic model for situation and threat assessment, *in Proceedings of the SPIE – The International Society for Optical Engineering*, Vol. 3080, pp. 171–182.

37. Pearce, A., Caelli, T. & Bischof, W. (1996) Claret: A new relational learning algorithm for interpretation in spatial domains, *in Proceedings of the Fourth International Conference on Control, Automation, Robotics and Vision (ICARV'96), Singapore*, pp. 650–654.

38. Pearce, A., Caelli, T. & Goss, S. (1996) *Aeronautical Parser for Spatio-Temporal Rules obtained by Machine Learning*, Technical report, Department of Computer Science, Curtin University and Aeronautical and Maritime Research Laboratories, DSTO.

39. Pearl, J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann.

40. Pignon, J.-P. & Suebe, N. (1992) A reactive mission planner for autonomous underwater vehicles, *in Proceedings of the Conference on Undersea Defence Technology*, pp. 585–590.

41. Pignon, J.-P., Guidet, Y. & Grabisch, M. (1992) Tactical decision support function for integrated on-board target intents analysis, *in Proceedings of the Conference on Undersea Defence Technology*, pp. 655–659.

42. Raff, S. J. (1991) The navy passive ranging problem, *Computers and Operations Research* **18**(7), 611–614.

43. Randell, D. A. & Cohn, A. G. (1992) Exploiting lattices in a theory of space and time, *Computers Math. Applic.* **23**(6–9), 459–476.

44. Rasmussen, J. (1983) Skills, rules and knowledge; signals, signs and symbols and other distinctions in human performance models, *IEEE Transactions on Systems, Man and Cybernetics* **13**(3), 257–266.

45. Rasmussen, J., Pejtersen, A. M. & Goodstein, L. P. (1994) *Cognitive Systems Engineering*, Wiley Series in Systems Engineering, John Wiley and Sons.

46. Ryder, J. M. & Zachary, W. W. (1991) Experimental validation of the attention switching component of the COGNET framework, *in Proceedings of the Annual Meeting of the Human Factors Society.* Also available from http://www.chiinc.com/.

47. Saito, T., Ortiz, C. & Loftin, R. B. (n.d.) On the acquisition and representation of procedural knowledge, http://www.vetl.uh.edu/KnowSys/arpk.html.

48. Sami, Y. & Vidal-Naquet, G. (1991) Formalisation of the behaviour of actors by coloured petri nets and some applications, *in Proceedings of the Conference on Parallel Architectures and Languages Europe, Lecture Notes in Computer Science, Vol. 506*, Springer.

49. Schank, R. C. (1972) Conceptual dependency, a theory for natural language understaning, *Cognitive Psychology.*

50. Schmidt, C. F., Goodson, J. L., Marsella, S. C. & Bresina, J. L. (1989) Reactive planning using a 'situation space', *in Proceedings of the Annual AI Systems in Government Conference*, IEEE Computer Society Press, Washington, DC, USA, pp. 50–55.

51. Travers, A. J. (1998) *Interval-based Qualitative Spatial Reasoning*, PhD thesis, Curtin University of Technology.

52. van Beek, P. & Manchak, D. W. (1996) The design and experimental analysis of algorithms for temporal reasoning, *Journal of Artificial Intelligence Research* **4**, 1–18.

53. Weiland, M. Z., Cooke, B. & Peterson, B. (1992) Designing and implementing decision aids for a complex environment using goal hierarchies, *in Proceedings of the Annual Meeting of the Human Factors Society*, Santa Monica, CA, USA, pp. 394–398.

54. Wohl, J. G., Alexandridis, M. G., Entin, E. E. & Deckert, J. C. (1985) Cognitive simulation of military decisionmaking, *in Proceedings of the International Conference on Cybernetics and Society*, New York, NY, USA, pp. 777–781.

55. Wohl, J., Serfaty, D., Entin, E., Deckert, J. & James, R. (1988) Human cognitive performance in antisubmarine warfare: Situation assessment and data fusion, *IEEE Transactions on Systems, Man and Cybernetics* **18**(5), 777–786.

56. Yampratoom, E. & Allen, J. F. (1993) Performance of temporal reasoning systems, *SIGART Bulletin* **4**(3), 26–29.

57. Zachary, W. W. (1988) Decision support systems : Designing to extend the cognitive limits, *in* M. Helander, ed., *Handbook of Human-Computer Interaction*, Elsevier Science, North-Holland, pp. 997–1030.

58. Zachary, W. W. (1989) A context-based model of attention switching in computer-human interaction domains, *in Proceedings of the Annual Meeting of the Human Factors Society*, pp. 286–290.

59. Zachary, W. W. (1996) Interface agents in complex systems, *in* C. A. Ntuen & E. H. Park, eds, *Human Interaction with Complex Systems : Conceptual Principles and Design Practice*, Kluwer Academic Publishers. Also available from http://www.chiinc.com/.

60. Zachary, W. W. & Weiland, M. Z. (1991) COGNET and BATON: An integrated approach to embedding user models in complex systems, *in Proceedings of the International Conference on Systems, Man and Cybernetics*, pp. 689–694.

61. Zachary, W. W., Ryder, J. M. & Hicinbothom, J. H. (to be published) Cognitive task analysis and modeling of decision making in complex environments, *in Decision making under streess: Implications for training and simulation*, American Psychological Association, Washington, DC, USA. Also available from http://www.chiinc.com/.

62. Zachary, W. W., Zubritzky, M. C. & Glenn, F. A. (1988) The development of the air anti-submarine warfare mission testbed as a tool for the development of operator models, *in Proceedings of the Annual Meeting of the Human Factors Society*, Santa Monica, CA, USA, pp. 1073–1077.

63. Zubritzky, M. C., Zachary, W. W. & Ryder, J. M. (1989) Constructing and applying cognitive models to mission management problems in air anti-submarine warfare, *in Proceedings of the Annual Meeting of the Human Factors Society*, Santa Monica, CA, USA, pp. 129–33.

DISTRIBUTION LIST

Representations and Processes in Decision Modelling

Stewart Greenhill, Svetha Venkatesh, Adrian Pearce and T.C. Ly

	Number of Copies
DEFENCE ORGANISATION	
S&T Program	
Chief Defence Scientist	1
FAS Science Policy	
AS Science Corporate Management	
Director General Science Policy Development	
Counsellor, Defence Science, London	Doc Data Sht
Counsellor, Defence Science, Washington	Doc Data Sht
Scientific Adviser to MRDC, Thailand	Doc Data Sht
Scientific Adviser Joint	1
Navy Scientific Adviser	Doc Data Sht
Scientific Adviser, Army	Doc Data Sht
Air Force Scientific Adviser	1
Director Trials	1
Aeronautical and Maritime Research Laboratory	
Director, Aeronautical and Maritime Research Laboratory	1
Chief of Maritime Operation Division	1
Research Leader Combat Information Systems	1
Head Submarine Combat System	1
Head Surface Ship Combat System	1
Thanh Chi Ly	1
Chris Davis	1
Simon Goss	1
DSTO Research Library and Archives	
Library Salisbury	1
Australian Archives	1
Library, MOD, HMAS Stirling	1
US Defense Technical Information Center	2
UK Defence Research Information Centre	2
Canada Defence Scientific Information Service	1
NZ Defence Information Centre	1
National Library of Australia	1

Capability Systems Staff

Director General Maritime Development	Doc Data Sht
Director General Aerospace Development	Doc Data Sht

Navy

SOSCI, Surface Combatants FEG Wartime Division, Maritime HQ, Garden Island	Doc Data Sht

Army

ABCA Standardisation Officer, Puckapunyal	4
SO(Science), DJFHQ(L), MILPO, Enoggera, Queensland 4057	Doc Data Sht

Knowledge Staff

Director General Command, Control, Communications and Computers	Doc Data Sht

Intelligence Program

DGSTA, Defence Intelligence Organisation	1
Manager, Information Centre, Defence Intelligence Organisation	1

Corporate Support Program

Library Manager, DLS-Canberra	1

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy Library	1
Head of Aerospace and Mechanical Engineering, ADFA	1
Deakin University Library, Serials Section (M List)	1
Hargrave Library, Monash University	Doc Data Sht
Librarian, Flinders University	1
Curtin Library, Curtin University	1

Co-authors School of Computing, Curtin University

Stewart Greenhill	1
Svetha Venkatesh	1
Adrian Pearce	1

OTHER ORGANISATIONS

NASA (Canberra)	1
AusInfo	1
State Library of South Australia	1

ABSTRACTING AND INFORMATION ORGANISATIONS

Library, Chemical Abstracts Reference Service	1

Engineering Societies Library, US	1
Materials Information, Cambridge Scientific Abstracts, US	1
Documents Librarian, The Center for Research Libraries, US	1

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK	1
Library – Exchange Desk, National Institute of Standards and Technology, US	1
National Aerospace Laboratory, Japan	1
National Aerospace Laboratory, Netherlands	1

SPARES

DSTO Salisbury Research Library	5

Total number of copies: 53

Page classification: UNCLASSIFIED

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. CAVEAT/PRIVACY MARKING

2. TITLE	3. SECURITY CLASSIFICATION
Representations and Processes in Decision Modelling	Document (U) Title (U) Abstract (U)
4. AUTHORS	5. CORPORATE AUTHOR
Stewart Greenhill, Svetha Venkatesh, Adrian Pearce and T.C. Ly	Aeronautical and Maritime Research Laboratory 506 Lorimer St, Fishermans Bend, Victoria, Australia 3207

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO–GD–0318	AR 012-146	General Document	March, 2002

8. FILE NUMBER	9. TASK NUMBER	10. SPONSOR	11. No OF PAGES	12. No OF REFS
M9505/17/163	DST 98/081	DSTO	61	63

13. URL OF ELECTRONIC VERSION	14. RELEASE AUTHORITY
http://www.dsto.defence.gov.au/corporate/reports/DSTO–GD–0318.pdf	Chief, Maritime Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved For Public Release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, SALISBURY, SOUTH AUSTRALIA 5108

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS

No Limitations

18. DEFTEST DESCRIPTORS

Situation awareness, Decision making, Knowledge representation, Knowledge acquisition

19. ABSTRACT

This report contains a survey by Curtin University on decision modelling which covers:

1. Our current understanding of how we make decisions, and points out our qualities, our weaknesses and the types of aids that could help us. Of note is the theory that people commit to options even though alternatives exist once a situation has been recognised.

2. Techniques useful for eliciting and representing knowledge about how experts make decisions.

3. What is situation assessment, and how others have tried to capture the process and use the captured information.

4. The different technologies that could be employed to represent the process of situation assessment.

This report represents the first step of a larger project to represent how submarine commanders assess situations.

GENERAL DOCUMENT DSTO-GD-0318 AR-012-146 MARCH 2002

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION
DSTO

AERONAUTICAL AND MARITIME RESEARCH LABORATORY
GPO BOX 4331 MELBOURNE VICTORIA 3001 AUSTRALIA
TELEPHONE (03) 9626 7000